

CONSTRUCTION, VIEWING AND PERCEPTION OF ANAMORPHOGRAMS

by

Lee-An

Robyn Gregory B.Sc.(Hons), Dip.Ed.

A Thesis submitted in fulfilment of the requirements for the degree of
Master of Science

in the

University of Tasmania

Dept of Mathematics

June, 1996

Abstract

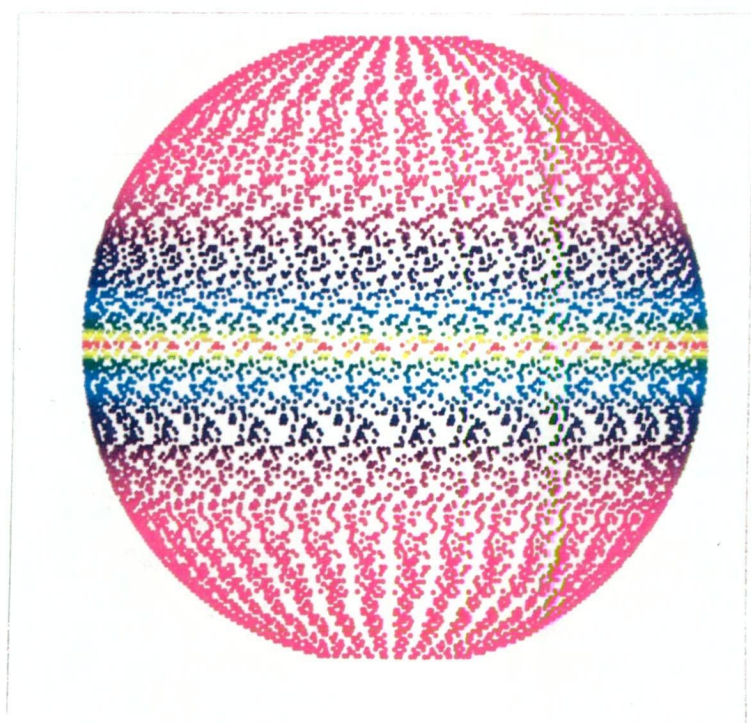
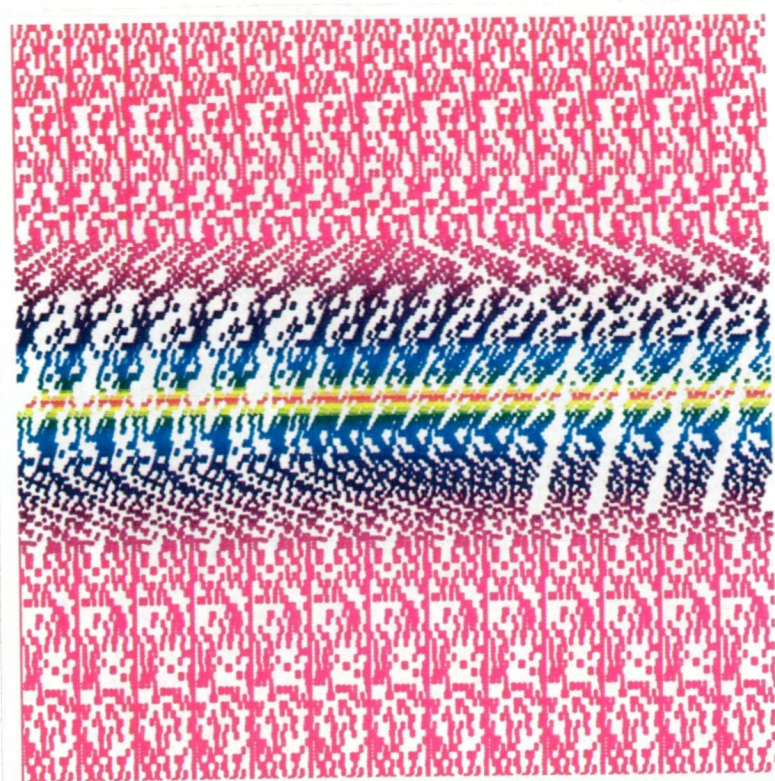
In this thesis we mathematically describe various means of constructing pictures which are camouflaged, and which can only be recognized when viewed in an unconventional, but prescribed manner. Such hidden, or distorted, pictures are referred to as *anamorphoses* or *anamorphograms*.

We begin by considering some physiology of the eye and examine some psycho-physical research in *Vision* and *Perception*, particularly related to *stereopsis*. Some confusion in this area is identified and discussed. The reader is guided through simple, but increasingly complex, experiments. These facilitate the viewing of, and the understanding of how we 'see' three-dimensional images in, *Single-Image Stereograms* (commercially referred to as 'Magic Eye' pictures). Construction of examples of these is one of our main foci. Various viewing techniques are described, and we identify some parallels between the psycho-physical analysis of stereopsis, and the likely results obtained by the reader in viewing the simple dot stereograms presented in our experiments.

We construct anamorphograms by applying a perspective drawing rule and some basic results of *Optics*. To be recognizable, our initial examples must be viewed monocularly from a prescribed viewpoint; or as a reflection in a given mirror; or wrapped around a given curved surface. Some of these examples are duplicated for binocular viewing in the form of *anaglyphs*, which allow the viewer to perceive a three-dimensional image.

Further anamorphograms, for binocular viewing, in the form of Single-Image Stereograms, are constructed. Some related geometry results are presented, culminating in the non-conventional representation of Single-Image Stereograms in terms of some results of *Projective Geometry*. We introduce the notion of a geometrical *stereoscope* which leads to the definition of a special *central collineation* and a consideration of a theorem related to fixed *conics*. Application of this result leads to the construction of a new Single-Image Stereogram of a sphere. This stereogram has special properties. We note the implications of the results of our introductory experimental section, for its viewing.

Finally, we present anamorphograms which are compositions of the preceding cases and we include transcripts, together with explanations, of the computer programs for creating all of our anamorphograms. These are written in *Mathematica*.



Contents

0.0	List of Figures	6
0.1	Statement	16
0.2	Acknowledgements	17
0.3	Notation	18
1	Introduction	20
2	Historical Introduction to Perspective Drawing, Anamorphic Art and Stereograms	23
2.1	Introducing Perspective Drawing	23
2.2	Introducing Anamorphic Art	25
2.3	The historical development of stereograms	26
3	Some Physiological and Perceptual Analysis of Stereopsis and the Viewing of Stereograms	31
3.1	The Eye and Vision	32
3.1.1	Some Physical and Psychological bases of Stereopsis .	37

3.1.2	Fusing matching retinal elements	50
3.1.3	Measuring receptive field size outside the eye	51
3.2	Mimicking Retinal Information for Stereopsis	55
3.2.1	Techniques for viewing stereograms	55
3.3	Building and viewing basic dot stereograms	57
3.3.1	Extension of dots to bigger rectangles	78
3.3.2	What can we deduce from these observations?	82
3.3.3	Interesting observation	84
3.4	Links between the viewing of stereograms and the physiolog- ical and psychophysical view of stereopsis	85
3.5	Some results of Luneburg	88
3.5.1	Experiment 1	89
3.5.2	Experiment 2	89
4	Creating Anamorphograms	97
4.1	Anamorphograms	97
4.2	Perspective Drawing	98
4.2.1	Parametric representation of a rectangular grid	99
4.2.2	Perspective drawing of a rectangular grid	100
4.2.3	Parametric representation of a cube	104
4.3	Slant Anamorphograms	108
4.4	Curved Surface Anamorphograms	109
4.4.1	Circular cylinder	111
4.4.2	Spherical surface	113
4.4.3	Conical surface	114

4.5	Mirror Anamorphograms	115
4.5.1	General mirror	117
4.5.2	Cylindrical mirror	119
4.5.3	Conical mirror	122
4.5.4	Spherical mirror	125
4.6	Binocular Examples	125
4.6.1	Anaglyphs	127
4.6.2	Revisiting Luneburg's experiment	129
5	The Construction of Single-Image Stereograms	135
5.1	An outline of a construction for Single-Image Stereograms . .	135
5.1.1	Constructing one row of dots on the xy -plane	136
5.1.2	Comparisons with some published papers	140
5.1.3	Extending to multiple rows	142
5.2	Single-Image Stereogram of an ellipsoid	144
5.2.1	Boundary considerations and further questions.	149
5.3	Single-Image Stereogram of cube	156
5.3.1	Which faces of the cube can we see?	156
5.3.2	Single-Image Stereogram representing a quadrilateral on one plane in space	159
5.3.3	Rotation of line segment onto xy -plane	160
5.3.4	Algorithm for one row of dots/line segment	161
5.3.5	z_{min} and z_{max}	162
5.3.6	Summary of steps for creating the entire Single-Image Stereogram of cube	163

5.4	Mixed Anamorphograms	171
5.4.1	Combining two perspective drawings, and a Single-Image Stereogram, of the same cube	171
5.4.2	Combining anaglyph and stereogram	171
5.4.3	Combining a stereogram and a cylindrical mirror anamorphogram	171
6	Single-Image Stereogram of a Special Sphere	176
6.1	The Stereoscope and some related theorems	178
6.2	Introducing Conics	185
6.2.1	Can we find a conic fixed under $\theta_n(l, r, D, n)$?	196
6.3	Constructing the Single-Image Stereogram of a Fixed Sphere	201
6.3.1	Method for finding an appropriate circle	201
6.3.2	Moving from one row, representing a circle, to many rows, representing a sphere.	203
6.3.3	Boundary problems	204
6.3.4	How do we find a suitable starting point?	207
6.3.5	Comments and observations	209
A	Matrix Manipulations	215
B	Program for creating perspective drawings	223
C	Program for Luneburg Experiment	228
D	Program for creating mirror anamorphograms	232
E	Program for creating Single-Image Stereogram of an ellipsoid	235

F Program for creating Single-Image Stereogram of a cube 248

**G Program for creating Single-Image Stereogram of a special
sphere 284**

Index 272

List of Figures

- 2.1 (a) Dürer’s woodcut; (b) Mapping p of object or scene to p'
on picture. 24
- 2.2 More distant objects appear smaller than nearer ones. 24
- 2.3 Two-picture Random Dot Stereogram (RDS) of Julesz 29
- 3.1 Horizontal section of an adult human right eye viewed from
above 33
- 3.2 Viewline; posterior nodal point is $\approx 17\text{mm}$ from the retina in
a normal eye 34
- 3.3 Refraction at the corneal surface 35
- 3.4 Man’s vertical and horizontal fields of vision 36
- 3.5 Fixating on a point, f 37
- 3.6 Corresponding points 38
- 3.7 Vieth-Muller circles for different fixation points 39
- 3.8 Corresponding points q_l and q_r 40
- 3.9 Corresponding points superimposed 40
- 3.10 Corresponding angles and arclengths 41
- 3.11 Disparity: difference between subtended angles 42
- 3.12 Disparity is the difference between the subtended angles 43
- 3.13 Overall temporal shift 44

3.14 Overall nasal shift	44
3.15 Fixating on far finger	45
3.16 Measuring disparity between s_1 and s_2	46
3.17 Measuring disparity - simplification using right triangles . . .	47
3.18 Aligning two stimuli movement along the appropriate Vieth- Muller circles	48
3.19 Distinction between 'depth' and 'distance' of a stimulus from the nodal points	48
3.20 Shaded region represents Panum's fusion area in one plane about the fixation point f	51
3.21 1mm on the retina = 3.5° of visual angle	52
3.22 Record of eye movements during free examination of the pho- tograph with both eyes.	53
3.23 Eye movements for subject reading Shakespeare.	54
3.24 Uncrossed eyes fixating beyond the stereogram	56
3.25 crossed eyes fixating in front of the stereogram	57
3.26 Focus behind the page	57
3.27 Perceived dots	58
3.28 Dots d_1 and d_2 ; perceived point p	59
3.29 Dots d_1 and d_2 , perceived point p ; dots d_3 and d_4 , perceived point p' ; dots d_5 and d_6 , perceived point q	60
3.30 Row of equally-spaced dots	61
3.31 Triangulation to give a row of equally-spaced dots	61
3.32 Row of unequally-spaced dots	62
3.33 Triangulation for every second dot to give a row of equally- spaced dots	63

3.34 Possible perceived dots	63
3.35 Possible perceived dots for two-picture dot stereogram	64
3.36 Guiding dot spacing is equal to s_d	65
3.37 Guiding dot spacing is twice s_d	65
3.38 Guiding dot spacing closer to s_d than to twice s_d	65
3.39 Guiding dot spacing closer to thrice s_d than to twice s_d . . .	66
3.40 Plane Π ; fixed line L ; points $s \in \Pi$ and $u \in L$	67
3.41 Obvious equally-spaced dots	69
3.42 Not so obvious equally-spaced dots	69
3.43 Guiding-dot spacing is that of consecutive dot spacing	70
3.44 Guiding-dot spacing given by horizontal spacing of vertical columns	70
3.45 Guiding-dot spacing much smaller than that given by hori- zontal spacing of vertical columns	71
3.46 \circ represents dots of vertical columns; \bullet represents random dots	72
3.47 Viewlines from each eye meet on a line	72
3.48 No obvious guiding dots	73
3.49 Two dots with smaller spacing from the rest	73
3.50 Geometry of the preceding case showing viewlines from each eye	74
3.51 Consecutive dot spacing varies across the row	74
3.52 Guiding dots 1: consecutive dot spacing varies across the row	75
3.53 Guiding dots 2: consecutive dot spacing varies across the row	75
3.54 \bullet row of unequally-spaced dots; \circ perceived dots: matching consecutive dots; \circ perceived dots: matching every second dot	76

3.55	Changes in horizontal dot spacing	77
3.56	Guiding dots force fixation on another plane for fusing dots .	78
3.57	Perceived picture relies on the viewing technique	79
3.58	Fuse using the 'uncrossed' technique	79
3.59	Cross-sectional view from above of the fused preceding figure	80
3.60	Circle spacing = s	81
3.61	Circle spacing < s	81
3.62	Circle spacing > s	81
3.63	two smallest circles with spacing much less than s	83
3.64	Cross-sectional view of geometry of the fused preceding figure from above	84
3.65	Horizontal lines to be viewed using 'uncrossed' technique . . .	85
3.66	Near point, n , for dot stereogram is outside the Vieth-Muller circle	86
3.67	(a) Correspondence of angles with dot-spacing; (b) Angle varies slightly for a particular dot-spacing	87
3.68	Vieth-Muller circle	90
3.69	Rotation of Vieth-Muller circle about the line $l \vee r$	91
3.70	pencils of red and green rays with the eyes as vertices	92
3.71	Lines of pencils intersect on Vieth-Muller circles	93
3.72	Lines of pencils intersect on Vieth-Muller circles	94
3.73	Cone of view and cone of fused lines	96
4.1	p' is image of p ; picture plane P ; viewpoint v	98
4.2	Figure F ; p' is image of p ; picture plane P : $y = 0$; viewpoint v	99
4.3	Rectangular grid	100

4.4	Rectangular grid of squares	101
4.5	Perspective drawing on the plane, $z = 0$, of rectangular grid .	102
4.6	Perspective drawing on the plane, $y = 0$, of rectangular grid .	103
4.7	Side views showing some viewlines from v to points, p , on our object figure	103
4.8	Cube of sidelength 2dim centered at the origin	105
4.9	Perspective drawings of cubes of similar dimensions, but dif- fering viewpoints.	107
4.10	Perspective drawing of cube on the plane $y = 0$	108
4.11	Slant anamorphic cube	109
4.12	Perspective drawing of cube on the xz plane; viewpoint $(0, -30, 0)$ 110	
4.13	Picture surface is a circular cylinder	111
4.14	Cross-sections viewed from above; near and far	112
4.15	Tangential viewlines place bounds on size of object	112
4.16	(a) Cube for near cylindrical picture surface ; (b) Cube for far cylindrical picture surface	113
4.17	(a) Cube for near spherical picture surface ; (b) Cube for far spherical picture surface	114
4.18	Two picture image points for circular cone	115
4.19	Viewpoint $(0, 0, 30)$, cone $x^2 + y^2 - 1/3z^2 = 0$, (a) inside top of conical picture surface (near), (b) outside far section of cone	116
4.20	Viewpoint $(0, -30, 30)$, cone $x^2 + y^2 - z^2 = 0$, (a) outside top of conical picture surface (near), (b) inside far section of the top section of cone	116
4.21	Angle of incidence = Angle of reflection	117
4.22	Viewer, image-object connections	118

4.23	Vector representation of triangles on plane of incidence. . . .	118
4.24	Surface reflections for near and far mirrors	120
4.25	Images of grid and cube for outside cylindrical mirror $x^2 + y^2 - 9 = 0$; viewpoint $(0, -30, 30)$	121
4.26	Viewpoint v , object p , normal N_m	122
4.27	Anamorphogram of grid for a conical mirror: (a) scale diagram; (b) correct scale, part of anamorphogram; (c) Image seen in mirror	123
4.28	Anamorphogram of cube for conical mirror: (a) correct scale, part of anamorphogram; (b) scale diagram	124
4.29	Viewpoint v , object p , normal N_m	125
4.30	Spherical mirror anamorphogram of the rectangular grid shown in the inset.	126
4.31	Spherical mirror anamorphogram of a cube	127
4.32	Perspective drawings of cube: green image for viewpoint $(2.875, -30, 0)$, red image for viewpoint $(-2.875, -30, 0)$	128
4.33	Anaglyph for cylindrical surface anamorphograms of a cube .	129
4.34	Anaglyph for a cylindrical mirror anamorphogram of a cube .	130
4.35	Viewpoints directly above l and r	131
4.36	Anamorphogram 1...red pencil through l and Anamorphogram 2...green pencil through r	132
4.37	Green viewpoint: $(2.875, 0, 10)$; red viewpoint: $(-2.875, 0, 10)$	133
4.38	Combined perspective drawings for both viewpoints	133
5.1	Sloping plane through the eyes intersecting a surface	136
5.2	Left eye: l ; right eye: r ; viewlines to points, p_i , on curve . . .	137
5.3	Adding random dots to the starting interval	138

5.4	Solid black regions are created in the interval to the left . . .	139
5.5	Large gaps are created in the interval to the right	139
5.6	Maeder's depth information lies between the two planes P_1 and P_2 parallel to the plane of the stereogram	140
5.7	Viewline from r cuts the curve at two points	142
5.8	Side view of sloping plane through the eyes	143
5.9	Relative positions of the y and y_i axes	144
5.10	Left eye: l , right eye: r ; stereogram at $y = d$; ellipsoid centered at $(0, a, 0)$	144
5.11	Outside front surface of the ellipsoid	145
5.12	Inside back surface of the ellipsoid	145
5.13	l, r, p are planar vectors	146
5.14	Rotation of axes through an angle of $\theta = \arctan[k/d]$	147
5.15	First and last dots for each eye	148
5.16	Matching dots and intersections with the ellipse of viewlines .	149
5.17	Tangent points on ellipsoid of plane through our eyes.	149
5.18	Single-Image Stereogram of the inside of a sphere with no background.	150
5.19	Tangents intersect behind the stereogram.	151
5.20	Single-Image Stereogram of the inside of a sphere.	152
5.21	Single-Image Stereogram of the outside of a sphere.	153
5.22	Single-Image Stereogram of the inside of an ellipsoid.	154
5.23	Single-Image Stereogram of the outside of an ellipsoid.	155
5.24	Left eye: l , right eye: r ; stereogram at $y = d$; cube with vertices 1,2,3,4,5,6,7,8	156

5.25	View looking down on cube, plane and viewpoint.	157
5.26	Cube and viewpoint	157
5.27	The angle between vectors	158
5.28	Sloping plane, Θ , through the eyes cuts the quadrilateral, Π'	159
5.29	Line segment which is the intersection of the sloping plane with Π'	160
5.30	Using our algorithm to find the dots	161
5.31	Finding maximum and minimum z using viewlines from mid- point of the eyes	162
5.32	Cube of dimension $2dim$, centered at the origin	163
5.33	Five possible k values for rows of dots representing different planes as we move across a row; three visible faces of the cube	164
5.34	Three visible faces of a cube; viewpoint: $(0, -30, 0)$; origin is at the centre of the page	166
5.35	Two visible faces of a cube; viewpoint: $(0, -30, 0)$; origin is at the centre of the page	168
5.36	One visible face of a cube; viewpoint: $(0, -30, 0)$; origin is at the centre of the page	170
5.37	Three visible faces of a cube; viewpoints: $(-2.875, -30, 0)$ and $(2.875, -30, 0)$	172
5.38	Three visible faces of a cube; viewpoints: $(-2.875, -30, 0)$ and $(2.875, -30, 0)$	173
5.39	Three visible faces of a cube: viewpoints $(-2.875, -30, 0)$ and $(2.875, -30, 0)$, mirrorsurface: $x^2 + y^2 = 25$	175
6.1	Any two non-parallel lines meet in a point and any two par- allel lines meet at an ideal point	176
6.2	Viewing the same dots using both the crossed and uncrossed techniques	178

6.3	A stereoscope presenting the image, x , of points a and μa . . .	179
6.4	Inverse stereoscope	179
6.5	4 is the harmonic conjugate of 3 with respect to 1 and 2 . . .	179
6.6	Stereoscope of the n^{th} image of a and μa	181
6.7	Homologous elements in two corresponding figures	182
6.8	Elementary maps	182
6.9	A perspectivity between points on L and L' , with centre l . . .	183
6.10	A perspectivity between lines through l and l' , with axis L . .	183
6.11	Projectivity between ranges on L and L''	184
6.12	Projectivity between pencils through l and l''	185
6.13	Conic with generating bases l and r	186
6.14	Triangles abc and $a'b'c'$ are perspective from p	187
6.15	Hexagon abc and $a'b'c'$; opposite sides are $a \vee b'$, $a' \vee b$, $b \vee c'$, $b' \vee c$, $a \vee c'$, $a' \vee c$	187
6.16	The points a, c and b are collinear	188
6.17	The points a, c and b are not collinear	189
6.18	The points p, p', q, q', s, s' represent the positions of corre- sponding pairs of dots of a stereogram	191
6.19	Singular conic: d' fixed by μ ; S is a line	192
6.20	(a) two fixed points and (b) one fixed point	192
6.21	$efrl$ is a 4-pt circuit: pairs of opposite sides are $e \vee f$ and $l \vee r$, $e \vee r$ and $l \vee f$, $e \vee l$ and $f \vee r$	193
6.22	p, v and θv are collinear; p is the centre and L is the axis . . .	195
6.23	x^{-n}, x^n and d are collinear; d is the centre and D is the axis .	195

6.24	The points q_1, q_2, q_3 are diagonal points of the 4-pt circuit, $p_1 p_2 p_3 p_4$	197
6.25	$H(d''d, x^n x^{-n})$	198
6.26	Tangents at p_2 and p_3 meet at b on $a \vee c$	198
6.27	The point q , is on the line containing the harmonic conjugate d'' of d with respect to a_2 and a_1	199
6.28	$d \vee p_1$ is a tangent and $\theta_n(p_1) = p_1$	200
6.29	The x -axis runs through the central row of the stereogram . .	202
6.30	Tangent point is the same for both circles C_{xy_k} and C_{xz} . . .	203
6.31	$a_{i+1} = u$	204
6.32	Dot for the right eye is to the right of u ; (b) An enlargement of the area of interest in (a)	205
6.33	Dots change direction; (b) An enlargement of the area of in- terest in (a)	206
6.34	Sections of circle which may not be visible	207
6.35	Starting point on the y_k axis for each k	208
6.36	$d = 30, u=8, \text{stepsize}=0.09, \text{AbsolutePointSize}=2$	210
6.37	$d = 30, u=8, \text{stepsize}=0.075, \text{AbsolutePointSize}=1$	211
6.38	$d = 30, u=6, \text{stepsize}=0.09, \text{AbsolutePointSize}=1$	212
6.39	$d = 30, u=6, \text{stepsize}=0.09, \text{AbsolutePointSize}=2$	213
A.1	Clockwise rotation	216
A.2	Square; translation, clockwise rotation, anti-clockwise rota- tion, scaling, xshear, yshear	219
A.3	Rotation of $\pi/5$ about the point $(1, 1.5)$	220

0.1 Statement

This thesis contains no material which has been accepted for the award of any other degree or diploma in any tertiary institution, and to the best of my knowledge and belief this thesis contains no material previously published or written by another person, except when due reference is made in the text of the thesis.

Robyn L. Gregory

Access Authority

This thesis is not to be made available for loan or copying for two years following the date this statement was signed. Following that time the thesis may be made available for loan and limited copying in accordance with the Copyright Act 1968.

Robyn L. Gregory 28/8/'96

0.2 Acknowledgements

I would like to thank Don Row whose expertise, and fresh teaching approach, in 'Projective Geometry' inspired me during my undergraduate years. His persistence in convincing me to undertake a Masters project has led to four enlightening years. Likewise his guidance and often perceptive criticism have been valuable; not to mention his responsibility in introducing me to the game of golf.

My very sincere thanks go also to Simon Wotherspoon whose early help with my *Mathematica* programming was invaluable; Kym Hill and Elizabeth Chelkowska for their help with Latex and advice on computer graphics packages; Peter Trotter for his encouragement and assistance in the final stages of preparation of this thesis; Michael Bulmer for his expert help with *Mathematica* and for the use of his program which made the typing of my bibliography far less onerous; my university colleagues for their support; and my students whose intense interest in my work made it seem worthwhile.

Most importantly, my heartfelt thanks to my entire family, but particularly to Norm, Adelle, Nellie and Gillan, who have been so supportive, loving and tolerant.

0.3 Notation

In general *points* are represented by lower case letters such as a and b . However, in some instances, if cyclic ordering is required, then the digits 1, 2, 3, 4, 5, 6, 7 or 8 are used.

Upper case letters represent *lines*.

$a \vee b$ represents a line through the distinct points a and b .

If the points x and y lie on the line $a \vee b$ ($x, y \in a \vee b$) then $x \vee y$ represents the same line. We write $a \vee b = x \vee y$ in this case.

$\overline{a \vee b}$ represents the line segment with endpoints a and b .

$|\overline{a \vee b}|$ represents the length of the line segment $\overline{a \vee b}$.

\overline{A} represents any line segment on the line A .

$A \wedge B$ represents the intersection point of the distinct lines A and B .

$(a \vee b) \wedge (c \vee d)$ represents the intersection point of the distinct lines $a \vee b$ and $c \vee d$.

$A \cap f(x)$ represents the intersection points of the line A and the curve $f(x)$.

In general, planes are represented by large Greek letters such as Π . Otherwise, a script letter has been used.

$a \vee B$ represents the plane through the point a , $\notin B$, and the line B .

qf represents the arclength for a minor arc with endpoints q and r on the circumference of a circle. (page 38)

$(q_r q_l)$ represents a point which is the superimposition of two points q_r and q_l so that $(q_r q_l)(f_r f_l)$ represents the arclength for a minor arc with endpoints $(q_r q_l)$ and $(f_r f_l)$ on the circumference of a circle. (page 39)

p_d represents the depth of the perceived point, p , behind a stereogram. p'_d represents the depth of a perceived point, p' , in front of the stereogram. (pages 58 and 59)

s represents the distance between two matching dots, one for each eye, of a dot stereogram. (page 59)

e represents the eye-spacing of the viewer. This is measured between the centres of the pupils. (page 59)

The distance d of the viewer from the stereogram is the perpendicular distance from the line through the nodal points of the eyes to the plane of the stereogram. (page 59)

s_d represents consecutive dot-spacing for a row of a stereogram. (page 64)

d_f represents the dotspacing of guiding dots. (page 66)

\uparrow represents the ideal point of the family of lines parallel to it in the extended Euclidean plane. (page 176)

x^{-1} represents the inverse image for a stereoscope. (page 178)

x^{-n} represents the n^{th} inverse image. (page 181)

$H(3; 12)$ represents the harmonic conjugate of the point 3 with respect to the points 1 and 2. (page 180)

$\overset{el}{L} \bar{\wedge} l$ represents an elementary map between the points on L and the lines through l . (page 182)

$L(p_1, p_2, p_3, \dots) \overset{l}{\bar{\wedge}} L'(p'_1, p'_2, p'_3, \dots)$ represents a perspectivity between the points on L and the points on L' with centre l . (page 183)

$L(p, q, r, \dots) \bar{\wedge} L''(p'', q'', r'')$ represents a projectivity between the ranges of L and L'' . Similarly, we may denote a projectivity, η , mapping a line of a pencil with vertex l onto a line of a pencil with vertex l'' by

$$\eta : l \bar{\wedge} l'' \text{ or } l \overset{\eta}{\bar{\wedge}} l''.$$

We have similar notation in the dual cases. (page 184)

$L \ni l$ denotes the fact that ' L is a line through the point l '. (page 185)

Chapter 1

Introduction

In embarking on this project, our aim was to complete a work which presented Mathematics as a valuable tool in describing an area with which it is not commonly linked.

Our experience led to the observation that many students need to ‘visualize’ a mathematical situation, if they are to fully understand it. This, coupled with a desire to present it in an enjoyable and enlightening manner to students, and mathematical laymen, was the impetus behind this choice of topic. We have endeavoured to demonstrate how our mathematical model fits our own experience with many ‘visual’ examples. Some common problems with the mathematics in physiological and psychological literature are raised. The existence of such problems is not surprising as usually the writers in such areas have no expertise in mathematics. Any criticism is not intended to be derogatory. Discrepancies, in defining terms, came to our attention whilst researching the many disciplines linked with this project. Initially, our reading was to enhance our own understanding. Our resulting confusion, in some areas, led to careful considerations. Hence this thesis is unusual in its ‘cross-disciplines’ nature.

Psychologists can study an observer’s perception of a visual situation. Often marked differences occur between the true physical situation and the interpretation of the viewer. This is evidenced by the well-documented existence of visual illusions and suggests that it is impossible to provide an exact mathematical description of what we perceive. In this thesis we demonstrate that exact mathematical description of some solid objects certainly allows the presentation of retinal information which facilitates the perception of the described objects. In our cases the most common factors described as

contributing to visual illusions, such as light, shade, colour and experience, are removed. Our results show that binocular disparity and convergence are major factors in the perception of depth.

In Chapter 2, we present an historical introduction to perspective drawing, anamorphic art and stereograms. Some vital definitions and ideas are introduced to enhance the understanding of more formal discussions presented in later chapters.

Chapter 3 looks briefly at the physical features of the eye and vision, and discusses in some detail the physical and psychological bases of stereopsis. As a means of achieving this there is an extensive review of the literature in these areas. Some inaccuracies, and imprecise definitions are discussed. This chapter also guides the reader through simple experiments with the aim of justifying the comments in the text. We include a description of possible viewing techniques for stereograms and then, painstakingly, discuss the building and viewing of basic dot stereograms. Some basic geometrical results are included and applied to a physical situation. We see that in some cases the geometry fits the actual physical, or perceptual, view. In others there is some discrepancy. We are able to draw parallels (of a non-geometrical nature!) between the way in which we ‘perceive’ dot stereograms, and the way in which stereopsis is described in the literature. When this amounts to linking the eye, brain and perception, such results cannot be conclusive.

The creation of anamorphograms is the main focus of Chapter 4. We define what we mean by an anamorphogram, and then formally discuss the mathematics involved in creating ‘distorted’ rectangular grids and ‘distorted’ cubes. For each example we have a prescribed viewing technique. This could include ‘viewing in a cylindrical mirror’ for which a piece of mirrored paper is included in the pocket inside the back cover of this thesis. In cases where a conical or spherical mirror is required, the reader will not easily be able to check our results. A rectangular grid was chosen as it can be co-ordinatized to allow the creation, by hand, of a distorted version of a picture. This technique is demonstrated in Hickin [11] and Gardner [6]. Initially, our anamorphograms are designed for monocular viewing. However, at the end of this chapter, we present anaglyphs of the cube examples. These are designed for binocular viewing, using ‘spectacles... red and green’

such as those included inside the back cover of this thesis. We are able to ‘see’ a three-dimensional cube, as distinct from a two-dimensional perspective drawing of one. We also apply our theory of slant anamorphograms to give an explanation of a perceptual view introduced in Chapter 3. The programs, written in *Mathematica*, for creating our anamorphograms are included in the Appendices B and D.

Chapter 5 focuses on a construction for Single-Image Stereograms, in general. Although these stereograms are also examples of anamorphograms, it was decided that their complexities warranted a chapter of their own. We also discuss the construction of two stereograms, each representing a particular case; an ellipsoid in space and a cube in space. Consideration is given to the limitations on our models for viewing purposes. Such limitations can be identified as a result of our discussion in Chapter 3. Comparisons are made between our examples and some of those published in recent research papers. We finish this chapter by including examples of anamorphograms which may be ‘seen’ by employing combinations of our various viewing techniques.

Again the programs, written in *Mathematica*, for both stereogram constructions are included with explanations, in Appendices F and E respectively.

Finally, in Chapter 6 we introduce some more ideas from the area of Projective Geometry. We define a ‘geometrical stereoscope’ and describe dot stereograms in terms of this. We prove various theorems related to special stereoscopes. In particular, we describe a central collineation which maps a conic onto itself. This result is applied to a particular example, to enable the construction of a Single-Image Stereogram with special properties. The program for this is included in Appendix G.

It must be acknowledged that although all our programs are easy to use, and give reasonable pictures, they are not yet in polished form. The programs are written in *Mathematica* (Wolfram Research, Inc.; ISBN 0-201-51507-5, 1991) and some of our results were tested using *The Geometer’s Sketchpad* package (Key Curriculum Press, Inc.; ISBN 1-55953-034-0, 1992).

Chapter 2

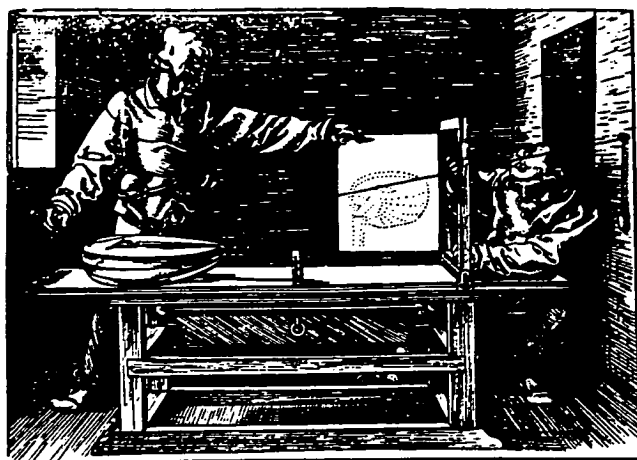
Historical Introduction to Perspective Drawing, Anamorphic Art and Stereograms

2.1 Introducing Perspective Drawing

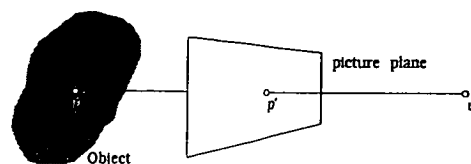
The word *perspective* is derived from the Latin words *per* (meaning by or through) and *spicere* (meaning to see or look). The main problem of perspective is:

How can 3-dimensions can be represented on a 2-dimensional page?

It was not until the Italian Renaissance that perspective was first developed successfully by Brunelleschi (1377-1446) and Alberti followed by suggesting that painting can be considered as a window through which we can see the visible world. Leonardo da Vinci (1452-1519) and Albrecht Dürer developed this idea into one where the artist is considered to be peering from a single point through a transparent screen at the object he wishes to paint. The points to be painted are located on the screen by finding the intersections with the screen of lines drawn from each point of the object to the eye. Many of Dürer's woodcuts, such as the one of Figure 2.1 (a) (From Cole [3, page 27]) show this technique. Figure 2.1 (b) simply demonstrates the basic idea. The eye of the artist is at v ; some distance in front of the artist we have a perpendicular plane (*the picture plane*). Now any point, p , of an object behind this plane is transferred to the point p' on the picture plane by finding where the line drawn from p to the eye intersects the picture



(a)



(b)

Figure 2.1: (a) Dürer's woodcut; (b) Mapping p of object or scene to p' on picture.

plane. We formalize this notion in section 4.2. Fortunately, the computer will help us create our pictures!

It is important to note that this rule of perspective was developed assuming that the artist was using one eye only. Provided that the viewer views from exactly the correct viewpoint then a very good illusion of depth is still created. We will consider binocular viewing in section 4.6.

Using this method of Dürer we can present objects as we see them and not necessarily as they are in fact. For example, the lines segments \bar{A} , \bar{B} , \bar{C} of Figure 2.2 are exactly similar in length, but according to this rule, their images on the picture plane are different lengths. This occurs because they are different distances from the viewer. More distant objects usually appear smaller to the viewer than similar objects which are placed nearer. Also

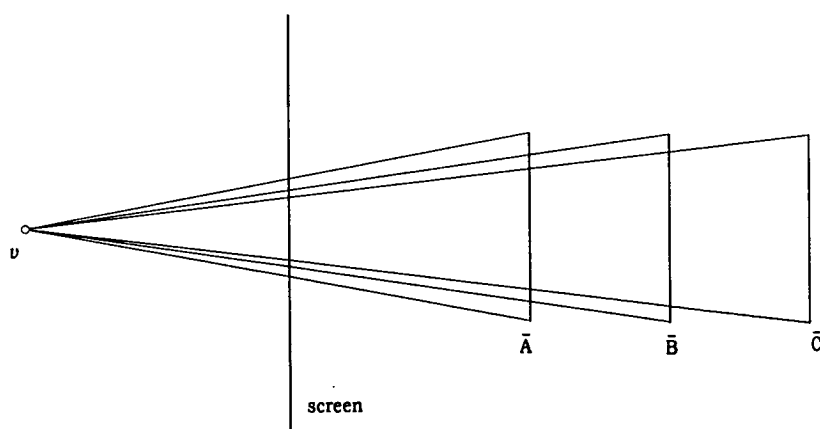


Figure 2.2: More distant objects appear smaller than nearer ones.

it was noted that in a picture, two parallel lines do not, in general, appear

parallel. For example, railway lines moving away from us appear to converge and meet at a point. Such occurrences led to the development of Projective Geometry by Desargues. This was as a result of the violation of the axioms of Euclidean Geometry, which include the notion that parallel lines never meet.

2.2 Introducing Anamorphic Art

Historically, anamorphic art developed during the Renaissance about the same time as perspective drawing. We begin by considering a brief history of traditional anamorphograms which were more commonly referred to as ‘anamorphoses’, although Kuchel [14] used the term ‘anamorphogram’. It is the perspective drawing rule which allows us to produce pictures which only look life-like when viewed with one eye, from a sharp angle near their side. Such anamorphoses we will call *slant anamorphograms*. The creation of these is sometimes referred to as using the rule of perspective in reverse. The likely explanation for this is mentioned in subsection 4.2.

Some of the earliest slant anamorphograms, which may be found in Leeman, Elffers and Schuyt [15, page 10], were drawn by Leonardo da Vinci. From the correct viewpoint, according to Leeman, Elffers and Schuyt [15], the picture ‘regains normal proportions and seems to rise and float free above the page’. The most famous of this type of slant viewing anamorphogram is Holbein’s double portrait *The Ambassadors*. A copy may be seen in Leeman, Elffers and Schuyt [15, pp 18-19] or in the theatrette in Edward St, The Glebe, Hobart, Tasmania. It is interesting to note that Hickin [11] points out that this painting of Holbein’s hangs in the London National Gallery in an unsuitable position to view the distortion correctly. This is also the case with respect the copy in Tasmania. We suggest that this is because very few people understand anamorphic art.

Anamorphoses supplied an ideal means of camouflaging themes which were to be kept secret, such as political messages, or obscene or erotic pictures. Some examples may be seen in Leeman, Elffers and Schuyt [15, page 13]. Anamorphic paintings, which must be viewed in cylindrical or conical mirrors, were fashionable toys in both Europe and China during the 17th and 18th Centuries. Salvador Dali created a set of erotic anamorphic paintings by simply looking into a cylindrical mirror while he painted on the surface

under it. Unlike Dali, we have the aid of a computer to allow us to apply our results and give us an accurate picture.

Another form of anamorphic art was the curved surface painting which became a problem for Fresco painters. They had to master not only painting on flat walls, but also, the technique for decorating curved walls and corners of vaults.

2.3 The historical development of stereograms

We now introduce the idea of stereopsis and some background to the conception of the Single-Image random dot stereograms. It is examples of these that are one of the main foci of this thesis.

Stereopsis is defined to be the perception of depth, and location of objects in space. It stems from the fact that the horizontal separation of our two eyes means that they view their surroundings from slightly differing vantage points.

Stereopsis enables a predator to penetrate the camouflage used by its prey, because monocular form perception is not a necessary pre-requisite for stereoscopic vision. For example, an insect disguised as a leaf may be invisible monocularly but stands out in depth when viewed stereoscopically.

Leonardo da Vinci pointed out that a sphere placed around the midline, and in front, of our eyes is seen differently by each eye: we see slightly further around the sphere on the left with the left eye, and on the right, with the right eye. The Principle of Stereopsis was first enunciated by Sir Charles Wheatstone in 1838 when he realized that the two projections of a cube onto the two retinas are different. This difference is evident in Figure 4.32. He invented the stereoscope which was present in almost every household for about half a century. In his original stereoscope, a viewer looked at two almost identical pictures in a box through two mirrors so that each eye saw only one picture. The small relative horizontal displacements of image points on the retinas lead to apparent differences in depth. In more recent stereoscopes than Wheatstones, the presentation of the appropriate image for each eye is achieved by two lenses. These are inclined so as to shift the images for each eye towards each other. This ensures the visual blending of the two images.

Early pioneers in photography turned to three dimensional photography as a means of adding extra realism to their pictures. By three dimensional photography we mean the use of a stereo camera. Such a camera has two lenses placed to mimick the positions of the human eyes. Consequently, two photographs of an object or scene may be taken simultaneously. A photograph of such a camera may be seen in Stereogram [24, page 36]. Stereoscopic photographs from this era documented many aspects of society from technology to art, to pornography, to wars and current events. An image viewed stereoscopically seems much sharper and brighter, and enables the viewing of greater detail (see examples in Stereogram [24, pages 36-41]).

In the 1950's anaglyph methods became popular for making comics where the pictures appeared to be three dimensional if viewed in a particular way. Such methods are experiencing a revival today in children's books. *Anaglyphs* are pictures that combine two pictures, one for each eye, into one picture. Traditionally, the picture for one eye is in red, and the one for the other eye, is green or blue. The result is viewed by wearing glasses with a red and a green 'lens', or filter, which is not a true lens, but a piece of coloured cellophane. Each eye can then only see the picture printed in the opposite colour to its lens. Again a three-dimensional image is perceived. There are many areas in which three dimensional techniques are used. Geographers study land formations using perceived three dimensional images resulting from stereo photographs taken from aeroplanes or weather balloons. Computer programs analyze photographs taken via satellites to construct surface maps of the earth and the moon and in chemistry, stereoscopic drawings of large complex molecules are made to help in understanding structure. In medicine three dimensional pictures are taken to study arteries. Stereo viewers can also be employed in detecting forgery of currency and the like.

Not only did Salvador Dali paint art to be viewed in cylindrical mirrors, he also painted examples of what have been termed *Stereograms*. Unlike the perspective art which came before, these paintings are especially created for the viewer to use both eyes. There are two paintings; one for each eye to view. As we will see in Chapter 3, the eyes combine with the brain to fuse the separate views for each eye stereoscopically to give a perceived three-dimensional image. Dali produced his stereo paintings by transferring images photographed with a stereo camera onto canvas. Examples of his work may be found in Stereograms [24, pp43-45]. For large paintings a

stereoscope such as Wheatstones must be used for viewing, but with practice, smaller examples can be fused with the naked eyes. If the distance between matching points on the two pictures is less than the viewer's eye-spacing, then the three-dimensional image will be seen behind the page. If the spacing is greater than eye-spacing, then the *crossed-eye* technique (see section 3.2.1) allows an image to be perceived which appears to float in space between the viewer and the painting.

While Salvador Dali's stereo art does not greatly camouflage the picture to be seen, we will create *Single-Image Random Dot Stereograms* where the intended image is completely hidden or disguised for monocular viewing.

One researcher who demonstrated the non-reliance on monocular cues for stereopsis is Dr Bela Julesz. He was introduced to aerial stereo viewing during his time as a radar engineer. In 1963 he invented an ingenious method for demonstrating stereopsis. He didn't want to show his test subjects stereo photographs which might have been used with a stereoscope, as he wanted to remove any possibility of subject recognition with monocular cues. He disguised his pictures by using a computer to draw pairs of random dot pictures that his test subjects viewed through a special viewer. For 'uncrossed viewing', provided that the matching dots for each eye are separated by a horizontal distance that is less than the observer's eye-spacing, then with practice, it is possible to view them without the aid of a viewer (for methods see subsection 3.2.1). By matching dots we mean the individual dots, one for each eye, which represent a single point of the object being viewed. These pictures of Julesz are commonly referred to as *Random-Dot Stereograms* or *RDSs*. An example of a Julesz-type random dot stereogram can be seen in Figure 2.3 (From Rock [25, page 62]). It can be seen that there are no obvious monocular cues to help with the recognition of the supposed image. When the two images are fused, a triangle is 'seen' apparently floating between the viewer and the page. It must be noted that a reader who is an inexperienced viewer of stereograms may have difficulty in fusing the images of Figure 2.3. Hopefully, after 'experiencing' this thesis perception may be possible. Although Julesz is credited with the invention of RDSs, it was pointed out by Dr C.W. Tyler in *Stereogram* [24, page 83] that in 1939, Boris Kompaneysky from the Russian Academy of Fine Arts published a random dot stereogram of the face of Venus.

The leaf room described in Tychsen [20, page 784], was also a fore-runner of the random dot stereogram. This is a two metre cubical box. Its inside surfaces are covered with leaves. Using binocular vision, the cubical shape of the box is easily perceived. This is not the case with monocular observation.

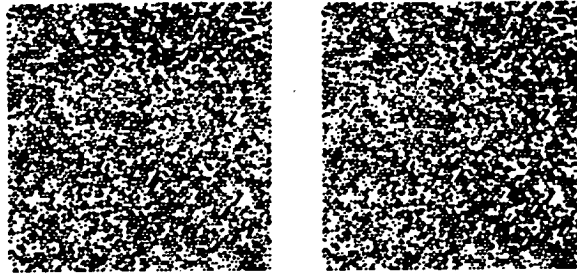


Figure 2.3: Two-picture Random Dot Stereogram (RDS) of Julesz

Julesz [13] describes the term *Cyclopean eye* which was first coined by Helmholtz to denote a hypothetical eye. This hypothetical eye was introduced by Hering to incorporate our two real eyes into a single entity. While Hering’s eye is described as being in the middle of our forehead (like that of the mythical Greek Cyclops), Julesz describes it as a central processing stage inside the brain that “sees” a single stereoscopic image given the appropriate stimuli in the two eyes. This process is commonly thought to be centered at the Striate Cortex, the terminus for the pathways of nerve fibres extending from corresponding retinal areas in the two eyes.

In 1983 as a result of considering the “wallpaper effect” C.W. Tyler [28] and [24] described a way of combining left and right eye information into one picture called an *Autostereogram*. This can be viewed easily without the aid of special glasses or a stereoscope. This effect was so-named because Dr David Brewster noticed that if the repetitive horizontal patterns in wallpaper and friezes are viewed in a particular way, then a depth impression is created. Further discussion of this effect appears in section 3.3.

It is examples of *Autostereograms* which we will consider in this thesis,

but we will refer to them as *Single-Image Stereograms*. This terminology has recently been used by Maeder [17]. Other examples are the commercial ones which are currently flooding the market such as those produced by N.E.Thing Enterprises in books such as 'The Magic Eye' [5]. Other research examples are those produced by Bar-Natan [1] and Terrell and Terrell [27]. In the creation of these stereograms we will use Dürer's technique for perspective drawing, however we combine the perspective drawings for two different viewpoints; one for each eye.

Chapter 3

Some Physiological and Perceptual Analysis of Stereopsis and the Viewing of Stereograms

This chapter is in response to the mathematics and computing papers relating to random dot stereograms which have been published during the three years that we have been working in this area. In these papers, and in earlier papers cited in these papers, many questions are left unanswered and many concepts are poorly defined. We could be forgiven for thinking that many of the stereograms published have been refined for easy viewing through trial and error. Very little information has been included relating the mathematics to the physiology of the human visual system, or to the perceptual view of stereopsis. This is surprising since random dot stereograms were conceived as a valuable tool in these areas for the testing of the presence of stereopsis in humans.

Statements such as the following examples abound:

“To aid in viewing a stereogram, the range of height values should not be too large in relation to the distance to the image plane. A maximum value of... is the largest one should try”, Maeder [17, page 53];

“The stereo effect is easier to obtain with a rather small disparity” or “A shorter repetition width produces a crossed disparity relative to the mean repetition width, while a longer one produces an uncrossed disparity”, Tyler and Clarke [28, page 185];

“The deviations from periodicity encode depth information”, “ There are too

few vertical strips to allow the random dot stereogram effect to be discerned easily", Dror Bar-Natan [1, page 70].

Such statements lack precision especially for a reader with no expertise in vision research. Whilst it is possible to do the mathematics and computing without such expertise, we felt it an essential part of this project to delve into the areas of physiology and perception. This gave us a greater understanding of the intricacies involved in creating satisfactory stereograms for viewing. We also wanted our stereograms to be as mathematically correct as possible. This does not always mean that the intended perceived image is the one which is most readily seen by the viewer. Possible reasons for this became apparent to us, and we hope will become apparent to any reader of this chapter.

The suggested 'reader experimental activities' in section 3.2, although laborious, are included to help the reader achieve some understanding, and also to facilitate the practice of various viewing techniques.

Before discussing the mathematics involved we will summarize some physiology and psychology of stereoscopic vision. This will aid in our understanding of how we might see Single-Image Stereograms. It will also help us make various assumptions and restrictions for our model. The evidence is a collation of results from many researchers. These results are by no means conclusive. Some inaccuracies and imprecise definitions will be discussed.

3.1 The Eye and Vision

The human eye possesses remarkable properties which allow most of us to view the outside world. In order to have some understanding of stereoscopic vision we will look at some of the eye's most vital physiological features. They will be by no means exhaustive. The positions of the cornea, iris, lens and retina can be seen on the diagram of the eye (see Figure 3.1). The most vital neural part of the eye is the retina. This, combined with the brain, contributes most of our ability to see. The collective function of the non-retinal parts of our eye is to keep a focused, clear image of the world, fixed on the retina. Each eye is held in its socket by six muscles that control its position. The cornea and the lens focus the light rays onto the back of the eye. The lens regulates focusing for near and far objects by becoming more, or less, convex. This changing of the shape of the lens is

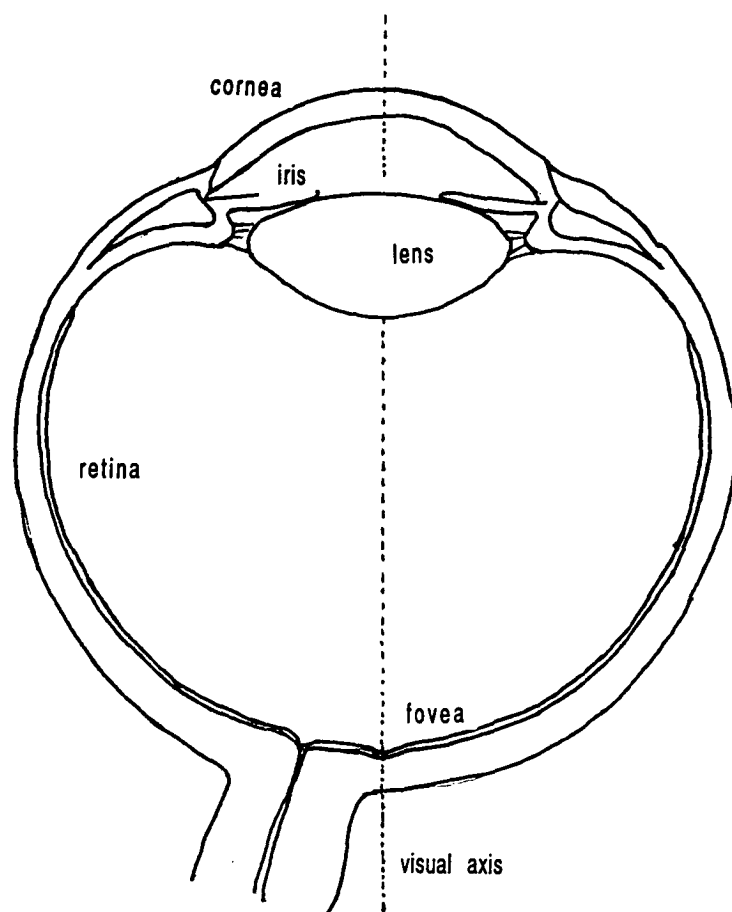


Figure 3.1: Horizontal section of an adult human right eye viewed from above

called *accommodation*.

The retina is part of the brain and is connected to it via a bundle of fibres called the optic nerve. It has the shape of a bowl and is about $1/4$ mm thick. It consists of three layers of nerve-cell bodies. The tier of cells at the back of the retina contains the light receptors, the rods and the cones (125 million of them). The rods are far more numerous than the cones, which are responsible for our ability to see fine detail and for colour vision. In the very centre of the retina where our fine-detail vision (*acuity*) is best, we have only cones. This rod-free area is called the *fovea*. It is a depression which occupies about 5° of arc or about 1.5mm on the retina where the cones are most densely packed. The very centre of the fovea is called the *foveola*. Functionally, the fovea is the position on the retina to which, by turning the eyeball, a person brings the image of whatever is of greatest interest in his *visual field* (illustrated in Figure 3.4).

Although the path of rays of light in the eye is in general quite complicated, there is one ray for each point source which travels in almost a straight line from its external source to the retina. It is in general a sufficient approximation to consider that all such lines cross inside the eye at one point called the *posterior nodal point*, situated towards the rear of the lens. In order to obtain the retinal image of any object, we will imagine a straight line drawn from each point on its surface to the nodal point and then extended to the retina. We will refer to this line as the *viewline* for each point (see Figure 3.2).

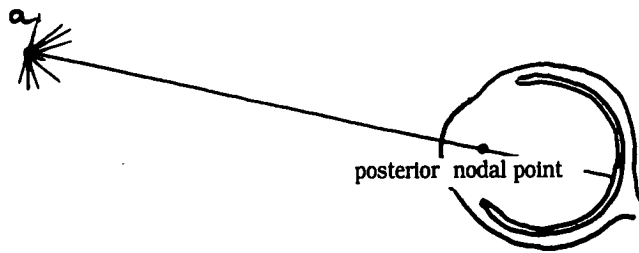


Figure 3.2: Viewline; posterior nodal point is ≈ 17 mm from the retina in a normal eye

The retina has a very pronounced curvature and covers the inside of the eyeball up to a region which is very close to the lens. It is because of this that we have a high degree of peripheral vision. Light which enters the eye

at a high angle (up to 104°) from the visual axis can still stimulate a point on the retina and so can be seen. This is due to consideration of refraction at the corneal surface (see Figure 3.3). We define the *visual axis* to be the

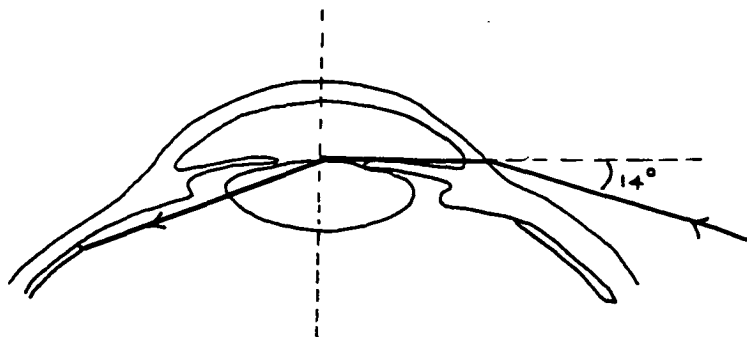


Figure 3.3: Refraction at the corneal surface

line through the foveola and the nodal point of the eye.

The vertical and horizontal fields of human vision are shown in Figure 3.4. The vertical range is about 140° and horizontally, it is about 150° for each eye. Man has binocular sight where the fields of vision for each eye overlap. This visual field is illustrated in Figure 3.4 by the darker shaded region. The smaller outer wedges represent the outer fringes of vision for each individual eye. Not surprisingly, peripheral vision is not as accurate as direct vision, as the rays need not fall on the retina near the fovea. When we fix our gaze or *fixate* on a given point, the eyes converge or diverge in order to bring the image of this point on to the fovea of each eye simultaneously. In this case, the point is seen only as a single point. The convergence or divergence movements where the eyes are turning simultaneously in opposite directions will be referred to as *vergence* movements. The amount of convergence varies inversely with the distance of the point. For very distant objects, such as stars, the viewlines are parallel and if we are viewing in this parallel way, the objects or points at other distances for which the convergence is not therefore correct, appear as two objects. This can be demonstrated by holding one finger at the level of your eyes while focusing on a distant point. We are aware of two blurred images of the finger. Similarly, if we focus on a nearer point than the finger, again two images can be seen. This effect is illustrated in Figure 3.15. The process where the brain allows us to see a single binocular image by combining two images, one for each eye, is called *fusion*. This terminology seems most appropriate as we shall see in our attempts to fuse two dots in section 3.3. It relies on appropriate eye

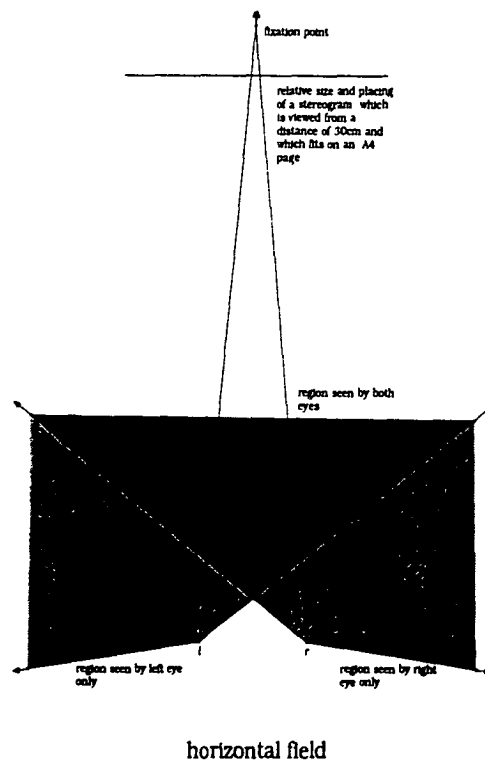
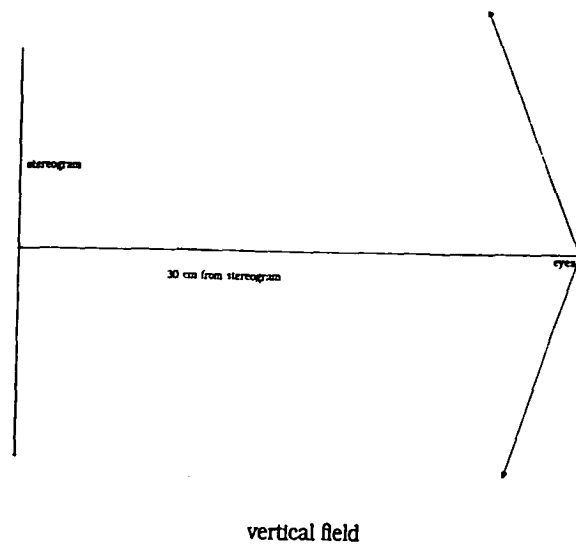


Figure 3.4: Man's vertical and horizontal fields of vision

vergence movements.

3.1.1 Some Physical and Psychological bases of Stereopsis

Stereopsis allows us to make relative depth judgements. It enables us to decide whether an object is nearer to us, or further away from us, than any other object within a region of space around a fixation point. Suppose a

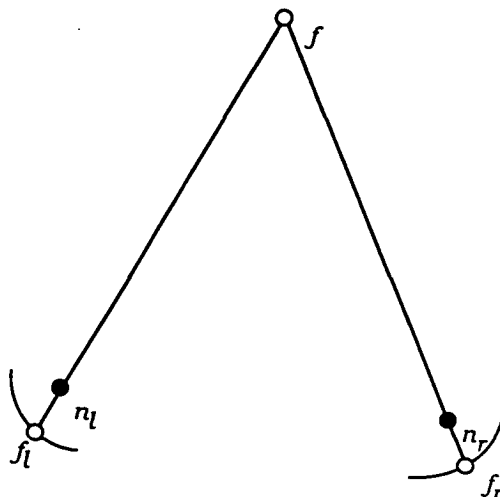


Figure 3.5: Fixating on a point, f

viewer fixates on a point, f , as illustrated in Figure 3.5. That is, he/she adjusts his/her eyes so that the images of f fall on the foveolas, f_l and f_r , of the eyes and the viewlines pass through the nodal points n_l and n_r . Now suppose that q is another point in space, within the peripheral vision region of the observer. Monocularly this point, q , will have an image on the retina that is to the right, left, above, or below the foveola. Binocularly, this point will project to either *corresponding* or *non-corresponding* points on the two retinas. According to Tychsen [20], *corresponding* points (such as q_l and q_r of Figure 3.6) on the two retinas are the same horizontal and vertical distance from the two foveolas. The lack of correspondence of two such images is referred to as *disparity*. These notions of disparity and corresponding points are defined in a variety of imprecise ways in the literature. Their definitions appear to have a geometric basis which stems from a consideration of the *Vieth-Muller circle* or *geometric horopter*. The *Vieth-Muller horopter* for a particular fixation point, f , is defined to be the set of points lying on a circle,

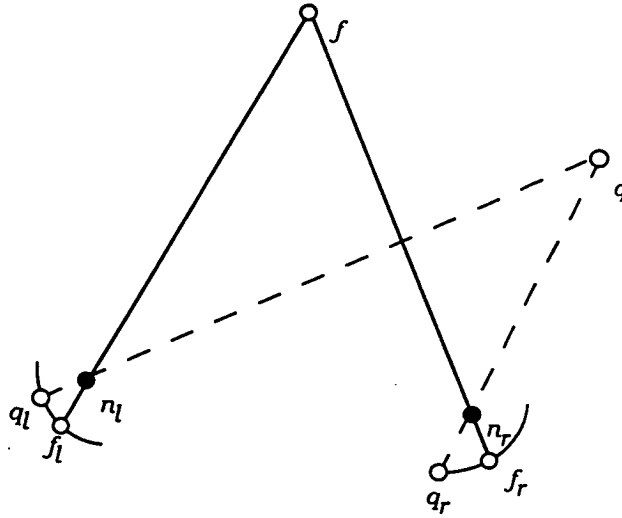


Figure 3.6: Corresponding points q_l and q_r : $q_l f_l = q_r f_r$

in the horizontal plane of the head, that passes through the nodal points of the two eyes and this point of fixation (see Figure 3.7). In most instances though, there is no mention of the horizontal plane through the head. In general, the *horopter* is variously described. Walonker and Feldon [30, pages 183–185] define their horopter to be the locus of all the points in external space that stimulate corresponding points on the retina. Perceptually, see Hubel [12, page 147], it is defined to be the set of points that are judged by the viewer to be the same distance away from him as the fixation point. In this latter description the term “distance” is ill-defined as it is a matter of judgement on the viewer’s part. Tychsen [20, page 779] suggests that “an object confined to the horopter will be seen as flat because it projects to corresponding retinal regions, causing zero horizontal disparity.”

Similarly, Julesz [13, page 165] says that all points that lie on the Vieth-Muller circle will have zero disparity, and that the Vieth-Muller circle is “an arc on the horopter surface, a surface of space having points which appear at the same depth”.

What would be the shape of this horopter surface? Could the notion of the Vieth-Muller circle be extended to a surface such as a torus obtained by rotating this circle about the line through the nodal points of the eyes? Luneburg [16] presented non-conclusive evidence to suggest that the answer to this question lies in a consideration of Hyperbolic geometry. We will present some of his ideas in section 3.5.

We will now consider some simple geometry of a circle as a means of understanding some of the definitions. As well we will point out some deficiencies in the use of this geometry in attempts to describe vision. Consider the Vieth-Muller circle for a fixation point f . *Note: This is sometimes referred to as the Vieth-Muller horopter but it is restricted to the plane through the fixation point and the nodal points of the eyes.* Any point q on this circle

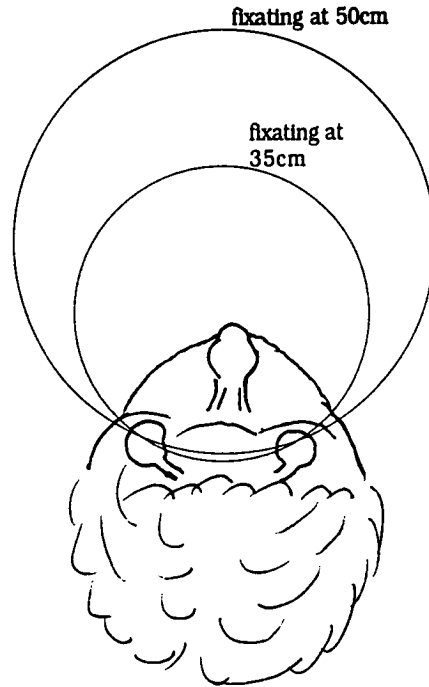


Figure 3.7: Vieth-Muller circles for different fixation points

will have image points, q_l and q_r , on the left and right retinas which are defined to be *binocularly corresponding points* (see Figure 3.8). Tychsen's [20] definition of corresponding points tells us that they are the same horizontal and vertical distance from their respective foveolas and that $q_l f_l = q_r f_r$. Walonker and Feldon [30] express this idea of corresponding points slightly differently. They suggest a superimposition of the two eyes as a pictorial representation of the imaginary cyclopean eye. In their case, the corresponding points coincide exactly (see Figure 3.9) so that arclength $(q_l q_r)(f_l f_r)$ (here the notation $(q_l q_r)$ represents the superimposed point (q_l on q_r) etc.) equals arclength $q_r f_r$ which also equals arclength $q_l f_l$. Now if we again examine Figure 3.8 and suppose that the angle subtended by the minor arc $n_r n_l$ at the point f is θ . This arc will also subtend an angle of θ at the point q on the same circle. (Angles in the same segment of a circle subtended by

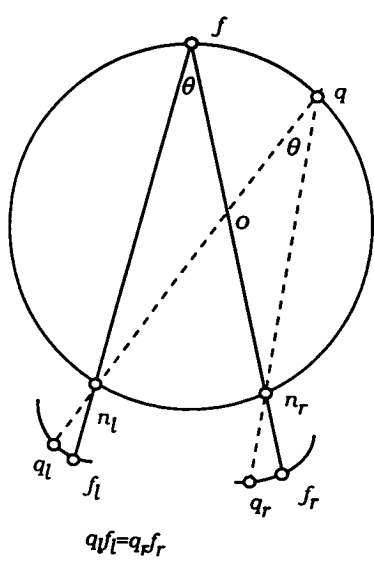


Figure 3.8: Corresponding points q_l and q_r

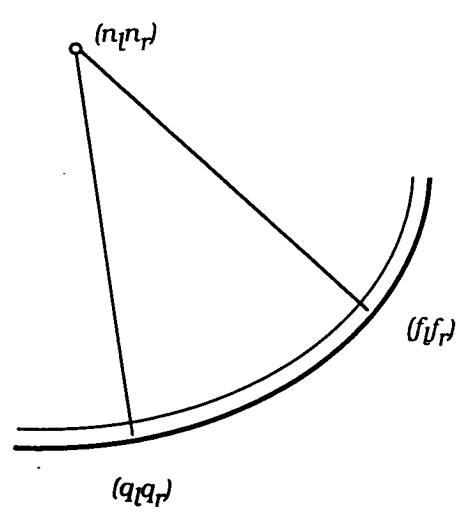


Figure 3.9: Corresponding points superimposed

the same arc are equal.) So we have the situation which we again depict in Figure 3.10 where triangles fon_l and qon_r are similar and so

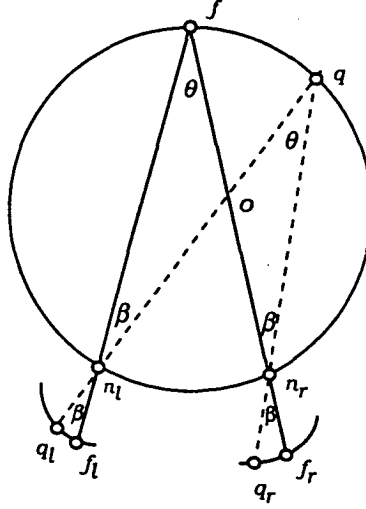


Figure 3.10: Corresponding angles and arclengths

$$\angle q_l n_l f_l = \angle q_r n_r f_r = \beta.$$

Next if we assume that the nodal points in each eye are in identical orientations with respect to the retinae then $\angle q_l n_l f_l = \angle q_r n_r f_r$ would imply that $\text{arclength } q_l f_l = \text{arclength } q_r f_r$.

Now, as we have seen, the lack of correspondence of two images on the retinas is defined as *disparity*. This disparity is described as a very powerful cue for depth perception. We will now look at this notion of disparity more closely. Again we examine our geometric horopter defined for a fixation point f (see Figure 3.11). Suppose a point s lies inside the major segment $n_l f n_r$ of the circle and that the angle subtended by the minor arc (chord $n_l n_r$) at s is ϕ . The disparity of the point s can be defined as the difference between the subtended angles at the points s and f . That is,

$$\text{disparity} = \phi - \theta.$$

Alternatively, if we let $\gamma = \angle s n_l f$ and $\beta = \angle f n_r s$, then

$$(i) \phi - \theta = \beta - \gamma$$

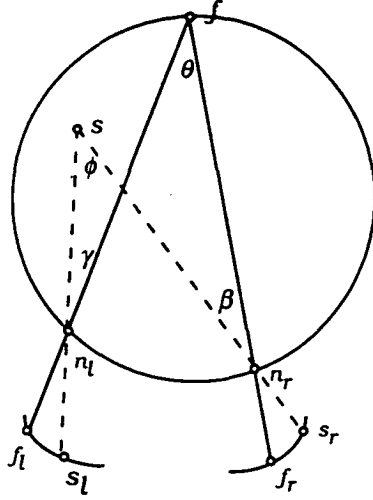


Figure 3.11: Disparity: difference between subtended angles

when s is outside the triangle $fn_l n_r$, but inside the left half of the Vieth-Muller circle,

$$(ii) \phi - \theta = \beta + \gamma$$

when s is inside the triangle $fn_l n_r$ and

$$(iii) \phi - \theta = \gamma - \beta$$

when s is outside the triangle $fn_l n_r$, but inside the right half of the Vieth-Muller circle.

Now the angles γ and β correspond respectively to distances $f_l s_l$ and $f_r s_r$ along the retina in each eye. These distances can be expressed in terms of arcminutes where

$$1 \text{ degree} = 60 \text{ arcminutes} = 3600 \text{ arcseconds}.$$

Suppose t is a point outside our circle but within the region of binocular vision of the viewer (see Figure 3.12). We have an analogous situation to that above, where again

$$\text{disparity} = \phi - \theta$$

and we have similar alternatives to those above for s in terms of β and γ . It must be noted that using this definition will give negative disparity in cases when t is outside the Vieth-Muller circle. Now if we examine Figures 3.11

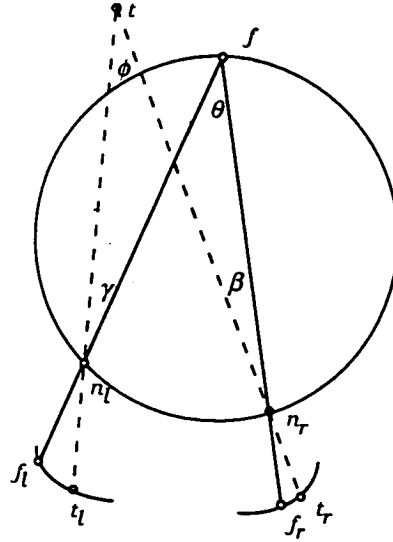


Figure 3.12: disparity = difference between subtended angles

and 3.12, we note that the shifts along the retinas relative to the foveolas, of any image points, can be either temporal (towards the temple) as in f_r to t_r (figure 3.12) or nasal (towards the nose) as in f_l to t_l (Figure 3.12).

Perceptually a point such as s (Figure 3.11) inside the horopter, is described as being a *near* point to the viewer in the sense that the viewer perceives that it is closer to him/her than the fixation point, f . According to Tychsen [20, page 780], such near points are described as giving rise to *crossed* disparities which are further characterized by a greater relative temporal shift of one of the retinal images. Conversely, a point such as t (Figure 3.12) outside the horopter, is described as a *far* point. Such far points give rise to *uncrossed* disparities which are characterized by a greater relative nasal shift of one of the retinal images.

If we geometrically examine the relative retinal shifts in each of the near and far point cases, then the above claims seem justified see Figure 3.13 and Figure 3.14, except in the cases where s and t are close to the central axis through the midpoint of the eyes (see Figure 3.13 (b) and Figure 3.14 (b)). In the latter cases, the retinal shift is the same for both eyes. Temporal in the near case and nasal in the far case. It is interesting to note that the reader demonstration given in Tychsen [20, page 780] appears incorrect. He attempts to justify the terminologies crossed and uncrossed disparity. He

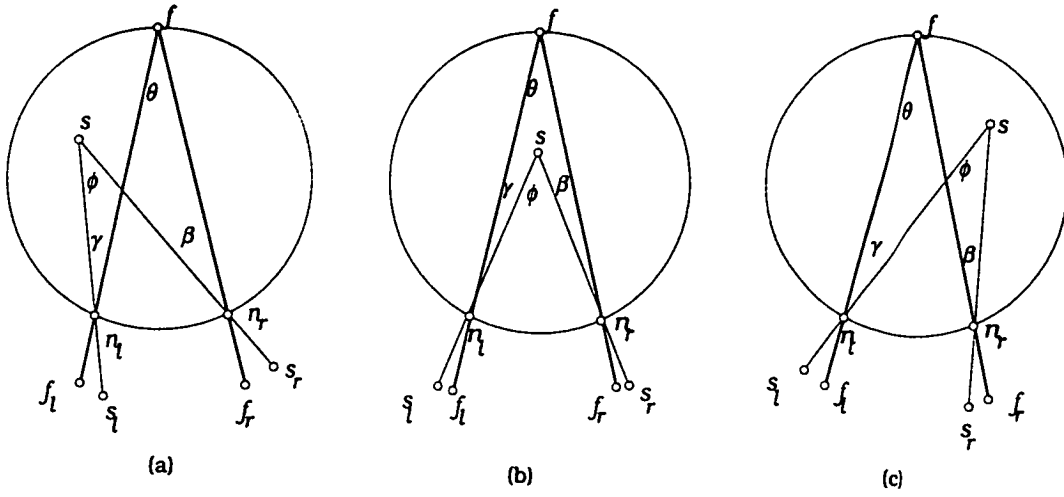


Figure 3.13: Overall temporal shift

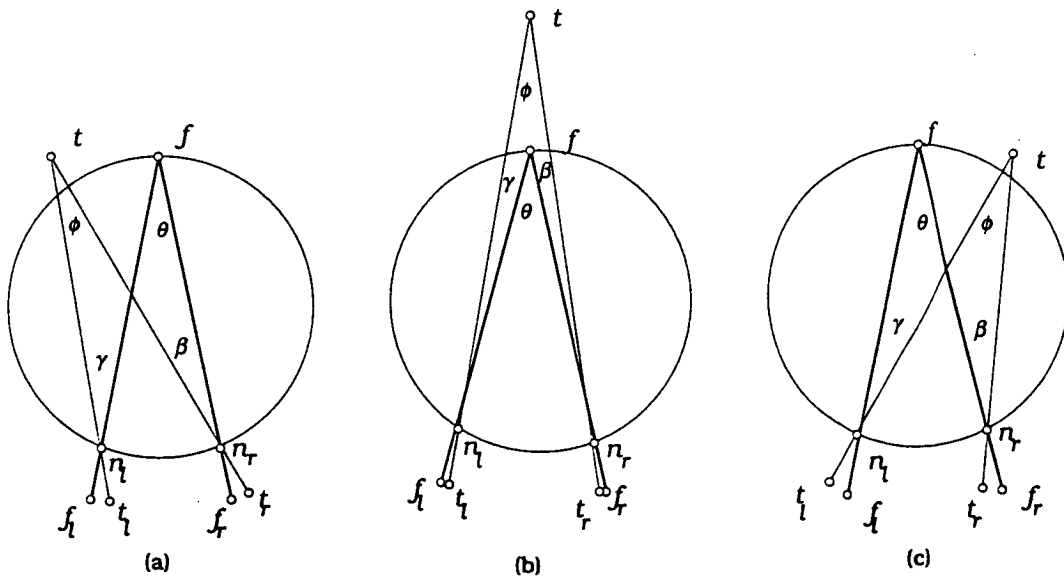


Figure 3.14: Overall nasal shift

incorrectly assumes that when a viewer is fixating on a distant point that the right-hand image of a nearer *double or diplopic* point is the right eye's image. Evidenced by his comment:

" With the right eye viewing monocularly (the left eye is momentarily shut) the image of the nearer finger is displaced to the left."

In reality the left-hand image of the double image is the one seen by the right eye. When the left eye is momentarily shut, the image seen by the right eye stays in the same position and is not displaced. This can be demonstrated by the following experiment. Basically it is Tychsen's experiment and diagram, but the appropriate corrections are made.

If we hold two fingers at different distances in front of our eyes, then when the more distant finger is fixated, the nearer finger is seen in crossed disparity as a double image (see Figure 3.15). Draw a coloured spot on the nearer

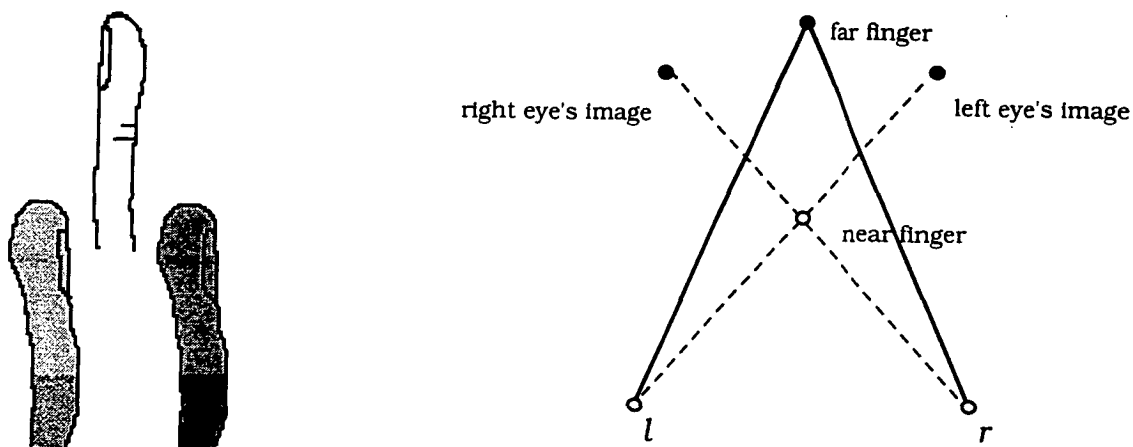


Figure 3.15: Fixating on far finger

finger. Place this spot in a position so that when it is viewed by each eye separately, only the right eye can see the spot. Now if we again fixate on the far finger and concentrate on the double image of the nearer finger, we can see that the image seen by the right eye is the left-hand image which has the coloured spot. This same exercise can be repeated with the nearer finger being the object of fixation. Then the more distant finger is seen in uncrossed disparity and this time the right eye does see the right-hand image.

It would seem that the most appropriate justification of the terms *crossed*

and *uncrossed* would be a consideration of the direction of the relative rotation of the eyes if they move from the point of fixation to fixate on the object presenting the original double image. In the near point case, this relative rotation would be nasal and in the far point case, temporal. For example, s_r in Figure 3.13, shifting from s_r to the foveola f_r would require a rotation of the eyes away from the nose. This we could describe as *uncrossing* our eyes.

It is interesting to note that in his discussion of near and far points, Hubel [12, pages 146-147] considers only the cases, (b), of Figure 3.14 and Figure 3.13. This may be justified as for this region about our visual axis, our acuity is greatest.

Suppose we want to measure the disparity between the stimuli s_1 and s_2 of Figure 3.16.

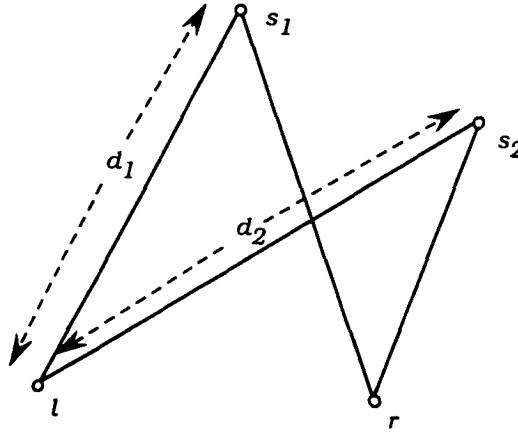


Figure 3.16: Measuring disparity between s_1 and s_2

Tychsen [20, page 781] suggests that we can simplify our calculations of disparity by considering right-angled triangles as shown in Figure 3.17. The two stimuli, whose disparity we are measuring, are aligned in front of one eye as shown. By measuring the distances d'_1 and d'_2 , we can easily calculate their disparity if we know the interpupillary distance (eye-spacing, e) of the viewer. That is

$$disparity = \phi - \theta = \arctan(e/d'_2) - \arctan(e/d'_1) = \beta.$$

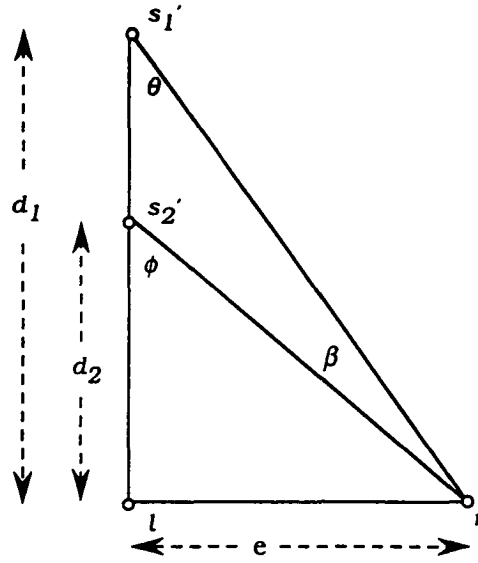


Figure 3.17: Measuring disparity - simplification using right triangles

In general this simplification seems inappropriate as in aligning two non-aligned stimuli, the angles ϕ and θ are preserved only if the movement of s_1 to s'_1 is along the Vieth-Muller circle for s_1 and similarly, for s_2 , as shown in Figure 3.18. Practically this does not seem helpful. Although we could measure d_1 and d_2 with a ruler or measuring tape, we have no simple way of finding d'_1 and d'_2 . If we were to approximate the d'_1 and d'_2 by the measurements of d_1 and d_2 respectively, then according to the angular definition of disparity, large errors occur.

However, if disparity is calculated by using the depth of the two stimuli, then there is no problem. By *depth*, denoted by d_p , we mean the perpendicular distance between a line through the nodal points of the eyes and a line through the stimulus point which is parallel to the line through the eyes as shown in Figure 3.19. In all the literature which is related to vision research and the physiology of the eyes there seems to be confusion between the two quantities 'depth' and 'distance'. Neither is well-defined. One possible definition of the *distance* of an object from the viewer is its distance from the midpoint of the viewer's eyes. This is denoted by d_m in Figure 3.19. Here m represents the midpoint of the eyes and again e denotes the eye-spacing. Luneburg [16, page 27] mentions that if a number of marks are arranged at equal distances on a Vieth-Muller circle, then a viewer with fixed head position will perceive these marks as though they are arranged on a circle with the observer as the centre. Later he specifies that the

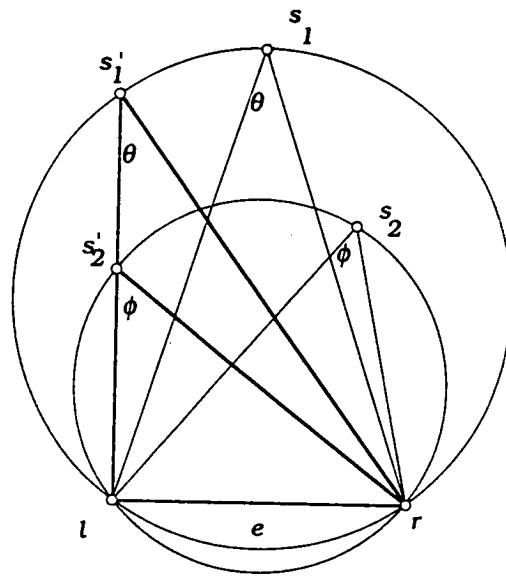


Figure 3.18: Aligning two stimuli movement along the appropriate Vieth-Muller circles

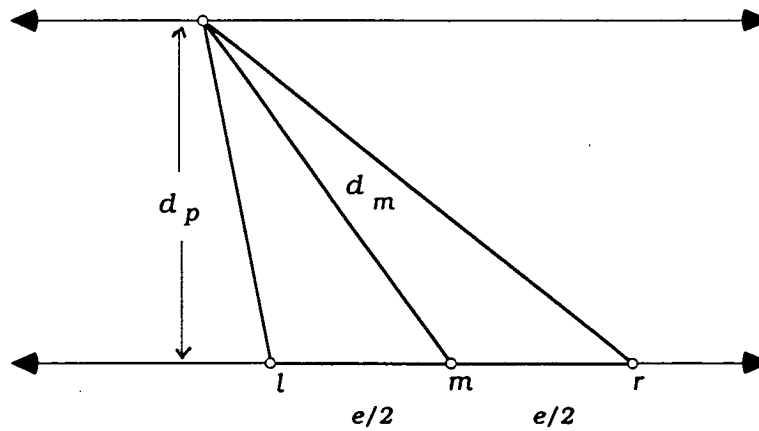


Figure 3.19: Distinction between 'depth' and 'distance' of a stimulus from the nodal points

centre of this interpreted circle is m , the midpoint of the eyes. That is, the viewer's interpretation differs from the physical reality. The reader may try this experiment using Figure 3.68 of section 3.5.

In the special case where the object lies on the central axis through the eyes its depth is equal to its distance.

It must be noted that the above definitions and discussion relate to the theoretical geometric circle (or horopter in one plane). Practically it has been shown by researchers that by using the perceptual definition of the horopter (*that is, the set of points that are judged by the viewer to be the same distance away from him as the fixation point*), some variation from the theoretical occurs. Luneburg's experiment has already been outlined. Another practical method for plotting the horopter for a particular fixation point is described in Tychsen [20, page 775]. This horopter is slightly flatter than the geometric horopter. Its deviation being attributed to complex neural and optical factors.

It appears that discussion and measurements of disparity in the literature relate to the geometric properties of the circle which we have discussed. This is regardless of the fact that the so-called perceptual horopter, with less curvature, would not now pass through the nodal points of the eyes. Since our eyes and brain are extremely sensitive to miniscule changes in distances along the retinae, any approximations made would seriously affect the credibility of the results.

The precise measurement of disparity is not important to this discussion. It is the concepts that are important if we are to develop some understanding of the three-dimensional effect we see in stereograms where we are presented with *matching* elements with changing horizontal disparity. Such elements, one for each eye, are representative of a single point of the object being viewed.

At this stage we can summarize by saying that when an observer fixates on a point of an object, the images of this point fall on the foveolas of the eyes. Any point in the field of view which the observer judges as being nearer than the fixation point, may have image points on the retinas with crossed disparity. This depends whether the actual geometry fits the perception of the viewer. Any point in the field of view which the viewer judges as being

further away than the fixation point may have image points with uncrossed disparity. Any point which the viewer judges as being the same distance away as the fixation point may have image points with zero disparity. Such points with zero disparity are described as corresponding points. By far the simplest way to judge whether a point is nearer or further away than some fixation point, is to measure in terms of depth. 'Depth' is much easier to define than 'distance', as it is the same for both eyes.

3.1.2 Fusing matching retinal elements

The fusion of two matching retinal elements to obtain a single binocular image is one of the essential aspects of stereopsis. The left and right retinal images of points on the horopter are "fused" for single vision and, by definition, have no disparity. However there is a region of single vision surrounding the horopter that does give *disparate* (non-zero disparity) image points on the retinae. This is known as *Panum's Fusional Space*. The perception of depth is dependent on the fusion of disparate retinal elements which are images of points within Panum's space. This space has a corresponding region on the retina within which image points must fall. This is called *Panum's area*. The horizontal extent of Panum's area varies according to different researchers. Originally Panum gave its size as representing a visual angle of approximately 6 minutes of arc. More recently its size has been extended. According to Tytchen [20], Panum's area is not of fixed size, but depends on whether the target is stationary or moving, and whether the eyes are stationary or moving. Hubel [12], says that it is equivalent to a visual angle of 2° , or about 0.6mm on the retina. It must also be noted that Piantanida [23] suggests that although the disparity of matching points can be increased to 2° or more, with fusion being maintained, refusion can only occur within much smaller disparities. A rough cross-sectional diagram representing the idea behind Panum's fusional space is shown in Figure 3.20. Here if f is the fixation point then p would present images on the retinas which can be fused provided

$$|\theta - \phi| \leq 2^\circ$$

or alternatively,

$$\theta - 2 \leq \phi \leq \theta + 2.$$

The point q for which

$$|\theta - \alpha| > 2^\circ,$$

would be seen as a double image. The diagram presented by Walonker and Feldon [30, page 186] is more specific about the shape of Panum's area. The

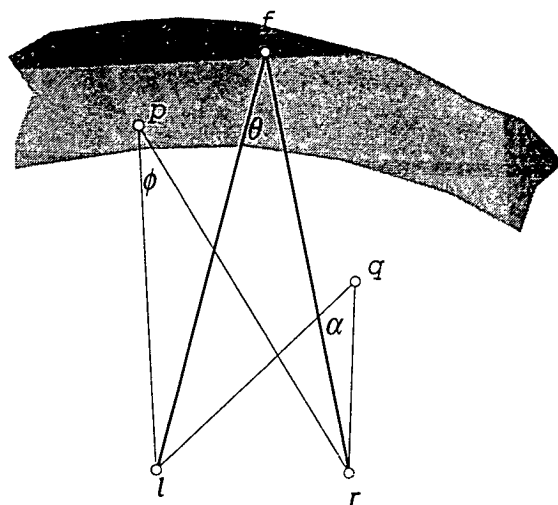


Figure 3.20: Shaded region represents Panum's fusion area in one plane about the fixation point f .

reader can roughly demonstrate the existence of this fusion space by carrying out the experiment of fixating on a far pencil point. Next raise a second pencil so that it is between the eyes and the far pencil. As mentioned previously in the case of a finger, unless this second pencil is in the vicinity of the fixated pencil, we see a double image of the nearer pencil. However, if we gradually move the closer pencil nearer to the far pencil, we become aware of the merging of the double image into one. We may conclude that a single image of the nearer pencil will be perceived when it lies within Panum's fusional space.

3.1.3 Measuring receptive field size outside the eye

Hubel [12] mentions a *receptive field* size measured outside the eye as shown in Figure 3.21. For the human eye one millimetre on the retina corresponds to 3.5° of visual angle. On a screen 1.5 metres away, 1mm on the retina

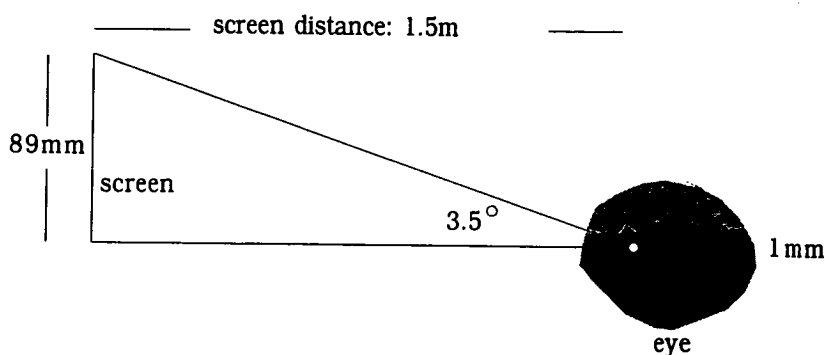


Figure 3.21: 1mm on the retina = 3.5° of visual angle

corresponds to 89mm. Figure 3.21 seems to give a specific case which approximates most situations in the same way that Tyschen's right-triangles of Figure 3.17 did.

Since the disparity represents a difference between two such angles, one for each eye, it is not obvious whether we can represent the disparity as a distance on a screen. We will investigate this possibility and make suggestions. *It should be noted that in general the term 'receptive field' refers to the specific receptors which feed into certain cells in the nervous system. In vision, it refers to a region on the retina. The size of such a region is hard to measure which is the reason that Hubel suggests the above measurement from outside of the eye.*

Before closing our discussion on the physiology of the eye we need to make mention of the *saccadic*, or jerking, movement of the eyes. When we explore our visual surroundings, rather than our eyes smoothly and continuously moving around the scene, they fixate on an object and hold that position for a brief period (Hubel [12] says approx. $1/2$ sec.) and then suddenly jump to a new position. They then fixate on this new target somewhere in the visual field. This new target may dominate in some way. During the jump, or saccade, the eyes move so rapidly that our visual system is unaware of the change. Researchers have monitored eye movements with the aid of a tiny mirror or coil of wire on a contact lens. In 1957, a Russian psychophysicist, A. L. Yarbus, monitored the eye movements as they explored various photographs with both eyes for one minute. One of his examples is shown in Figure 3.22 (From Adler [10, page 177]). Feldon and Burde [2, page 177]

report that Yarbus found: *In examining complex stimuli the eye foveates longest on elements with the relevant information, whereas many elements are never viewed foveally.*

Apparently, when reading, our eye movements demonstrate a “staircase”




Figure 3.22: Record of eye movements during free examination of the photograph with both eyes.

pattern. A record of this pattern for a subject reading a Shakespearean sonnet is shown in Figure 3.23 (From Adler [10, page 177]). In this case, the movements consist of alternating saccades and periods of fixation (about 100 to 500 msec each). Each saccade moves the fovea about eight characters to the right. At the end of each line a large saccade to the beginning of the next line occurs. When our eyes fixate on a point they do not lock into position as we might expect. They make tiny movements called *microsaccades* which occur 2 to 3 times per second and are about 1 to 2 mins of arc in amplitude and in random directions. These microsaccades, together with a high frequency tremor, are vital for the viewing of a stationary scene as they prevent the fading of the retinal image.

We have included this information, as we consider the question:

How do our eyes behave when viewing random dot stereograms?

Given the homogeneous nature of the dots, we don't usually have any dominant features which would suggest the possibility of the type of saccadic movements described for pictures. It is possible that our eye movements


 вы, мой стих не блещет новизной.
 Разнообразьем перешен неожиданных.
 Не поискать ли мне тропы иной,
 Приемов новых, сочетаний странных?

Я повторяю прежнее опять.
 В одежде старой появаясь снова.
 И кажется, по имени назвать
 Меня в стихах любяще может слово.

Всё это оттого, что вновь и вновь
 Решаю я одну свою задачу:
 Я о тебе пишу, моя любовь,
 И то же сердце, те же силы трачу.

Всё то же солнце ходит надо мной.
 Но и оно не блещет новизной.

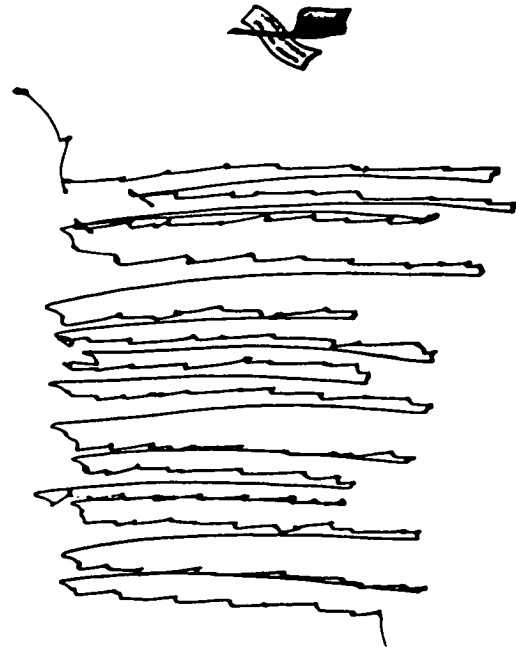


Figure 3.23: Eye movements for subject reading Shakespeare.

might demonstrate the type of 'staircase' pattern that we use for reading. We have horizontal rows of dots, just as we have horizontal rows of characters, when we read. This would be interesting to check experimentally, however, it is beyond the scope of this thesis. We can state, however, that the computer animation of the cube stereogram discussed in Chapter 5 leads to a vivid three-dimensional cube rotating in space. The speed with which the animation frames change would suggest that row by row scanning of all the dots is unlikely.

The question relating to eye movements arose because of a further question: *How do we perceive stereograms?*

In research papers, Bar-Natan [1], Mitchison and Westheimer [21] and McKee and Mitchison [19], dealing with random dot stereograms there is often some reference made to a 'fixation plane' behind the stereogram. Given that in stereograms we mimic retinal information to obtain stereopsis, this statement leads to an obvious connection between this 'fixation plane' and the 'horopter'. We now carefully explore the viewing of dot stereograms, to check this notion of a 'fixation plane'. Its possible relationship to the curved horopter, perceptual or geometric, will be examined.

3.2 Mimicking Retinal Information for Stereopsis

We will now look at how we can ‘trick’ our brains into believing that they are viewing a three-dimensional object. This can be done by mimicking the exact information that the retinas would receive if indeed, there was an object in the ‘perceived’ position. In order to appreciate the results of this section we need to master the techniques for viewing stereograms.

3.2.1 Techniques for viewing stereograms

It is important not to overstrain your eyes when trying the following techniques for viewing stereograms. If viewing correctly, your eyes will still feel quite relaxed. These techniques sometimes take a while to master. The aim is to force your right and left eyes to look at different places at the same time.

Technique 1: UNCROSSED or PARALLEL technique

This is the technique used when we want to view a stereogram where the perceived image (in depth) is behind the plane of the stereogram. Although it is sometimes referred to as the *parallel* technique, the viewlines from each eye are not parallel to one another once we have fixated at a suitable depth behind our page. To facilitate the fixation at a point behind the page, we sometimes stare into the distance to begin with, so that at this stage, the viewlines are parallel. The required situation can be represented schematically by the following Figure 3.24 which shows a cross-sectional view from above.

Methods for Uncrossed technique

Note: These same techniques can be used for two or more dots. If used for two guiding dots for a larger picture, then the aim is to maintain the state which allows you to see the three dots, and shift your attention to the larger stereogram.

(1) Look at the stereogram from a normal viewing distance but instead of focusing on the page, stare beyond it. After a few seconds, your eyes and brain will adjust to the correct fixation point for the stereogram and the image will ‘miraculously’ appear.

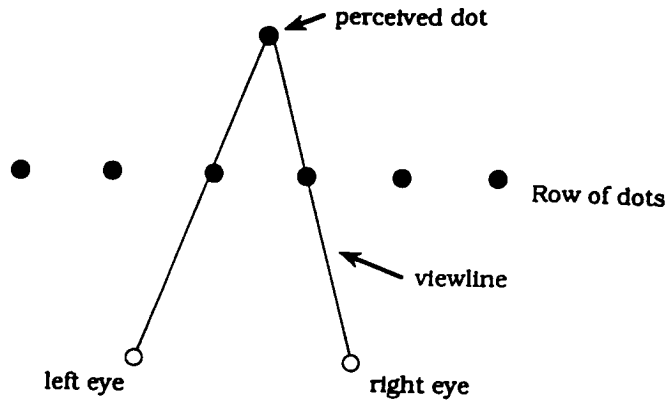


Figure 3.24: Uncrossed eyes fixating beyond the stereogram

(2) Place a transparent sheet of shiny material such as overhead projector film over the page of the stereogram. Sit in a position which allows you to see your reflection in this sheet. If you relax, and concentrate on this reflection which is behind the page, then in a few seconds the image will appear.

(3) Hold the page with the stereogram flat up against your face. Stare through this page as though you are viewing a distant object. Very slowly move the page away from your face. At some point, as you move the page away, the three dimensional image should appear.

(4) If none of the above methods work, then for viewing two dots or two-picture stereograms, place a piece of cardboard perpendicular to the page, and between, the two pictures or dots. The cardboard should be about the size of a postcard. If you stare blankly either side of the cardboard, the two pictures should fuse so that you see a single picture. You should then be able to slowly move the cardboard without losing the fused image.

Technique 2: CROSSED-EYES technique

This is the technique employed when we want to view a stereogram where the perceived image, in depth, is between the page of the stereogram and the viewer. It also has the advantage of being a possible method for viewing stereograms where the spacing between matching dots exceeds the eye-spacing of the viewer. It is called the *crossed-eyes* technique because it requires that as a viewer you cross your eyes; in a relaxed way. This can

be represented schematically by the following Figure 3.25 which shows a cross-sectional view from above.

Method for Crossed-eyes technique

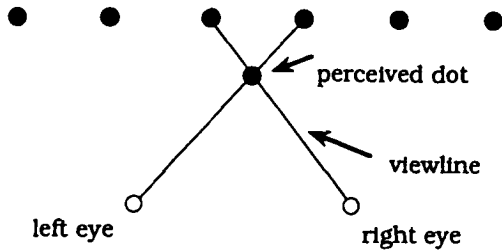


Figure 3.25: crossed eyes fixating in front of the stereogram

Hold up your finger between your eyes and the stereogram and stare at your finger. This will cause your eyes to cross. Hold your eyes as they are and shift your attention to the stereogram. You may need to move your finger back and forth until you find the point at which the three-dimensional image is clear.

We now return to our discussion of mimicking retinal information.

3.3 Building and viewing basic dot stereograms

Consider the two dots shown in Figure 3.26. Next concentrate on the dots

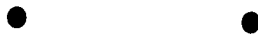


Figure 3.26: Focus behind the page

so that your left eye focuses on the left dot and your right eye focuses on the right dot. This means the lens of each eye will be accommodated for

its particular dot on the page, but the viewlines from each eye will meet behind the page. There are various techniques which can be used to achieve this eye positioning, referred to as the ‘uncrossed’ technique, which can be seen in subsection 3.2.1. The aim is to ‘fuse’ the two dots by adjusting the angles of your eyes until you have a stable image of three dots only. This often involves a stage of going through four blurry unstable dots first. When viewed correctly, the dots will appear as shown in Figure 3.27 where the ‘fused’ dot appears to be at the midpoint of the other two dots.

The description of this process as fusing the dots seems most appropriate as we do have the impression that we are superimposing one dot on top of the other. This impression is very marked if we attempt to fuse dots of differing size and shape, or dots of different colours. In these cases we are aware of what could be described as a shimmering effect. (The reader can easily check this out.) Our brain seems to alternately choose each of the different image dots as the perceived dot. This confusion is not surprising. In normal stereo viewing, the images of a single point on an object would be almost identical in size, shape and colour. Any minute differences could be attributed to the slightly different viewing positions of the eyes. In the discussion that follows we will be presenting identical dots to each eye. Diagrammatically, we can



Figure 3.27: Perceived dots

represent the situation as shown in Figure 3.28 where d_1 and d_2 represent the dots, l and r represent the nodal points of the eyes and p represents the perceived dot. The points l, r, d_1, d_2 and p are co-planar. Our perpendicular distance from the line $d_1 \vee d_2$, is d and p_d is the depth of the perceived point, p , behind the page. Unless otherwise stated, we assume that the viewer is standing *parallel* to the page of the dots. By this statement, we mean that the line through the nodal points, $l \vee r$, is parallel to the line through the dots, $d_1 \vee d_2$. The presence of three dots comes about because when the right eye is focused on d_2 , there will still be an image of the point d_1 on

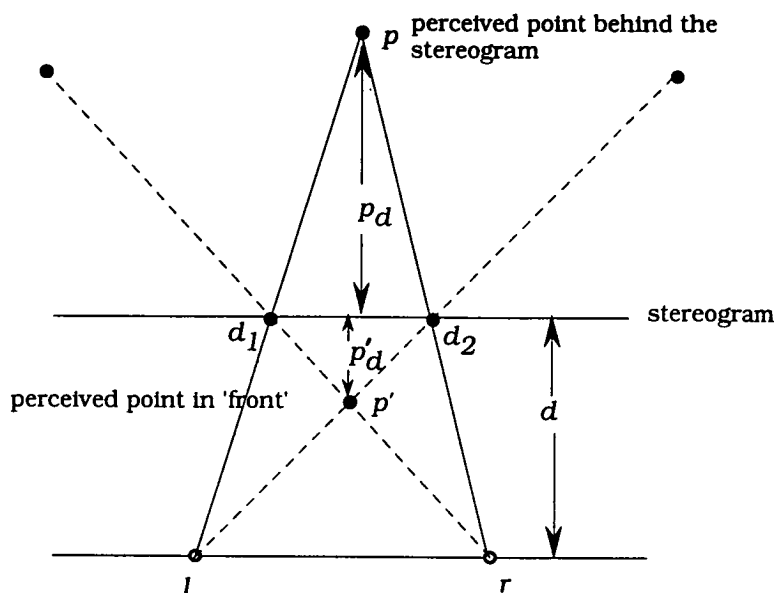


Figure 3.28: Dots d_1 and d_2 ; perceived point p

its retina (in peripheral vision). Similarly, the left eye focused on d_1 will still 'see' d_2 . Using geometry, the exact positions of these peripheral images can't be pinpointed in the same way that p can. Maybe they are 'projected' along the viewlines (dotted in Figure 3.28) to a depth similar to that of p ? (*uniocular projection* is mentioned by Luneburg [16])

If s represents the distance between dots d_1 and d_2 and e represents the distance between l and r (*eye-spacing*), then using similar triangle results we have

$$s/e = p_d/(p_d + d). \quad (3.1)$$

This is a significant result, as geometrically, if the dot spacing and viewing distance remain fixed, then the depth, p_d , of the perceived point p will be fixed.

We have an analogous result for "crossed-eyes" viewing. That is,

$$s/e = p'_d/(d - p'_d), \quad (3.2)$$

where in this case, $p'_d > 0$ represents the depth of the perceived point, p' , in 'front' of the row of dots including d_1 and d_2 . That is, between the viewer and the stereogram, and in the same plane as the dots and the eyes. Note: The eye-spacing, e , is fixed for any particular viewer. It does, however, vary

between individuals; for women the range is 55-62 mm; for men the range is 60-67 mm.

Theoretically, any stimulus on either viewline of Figure 3.28 would allow us to perceive the point p . The reader can demonstrate this for him/herself by drawing the two dots on different pieces of paper and then by moving one piece closer and closer along either of the viewlines, $p \vee d_1$ or $p \vee d_2$. It is to be noted that practically, this experiment demonstrates that our eyes cannot simultaneously be accommodated for stimuli that lie at vastly differing depths from the eyes. Consequently, the fusing of the images of the two dots becomes increasingly difficult as the difference in depth between the dots increases. If the presence of these dots as stimuli is to mimick a real situation, then it is sensible to keep our dots the same depth away from us. When a point of an object is fixated by our eyes it is obviously the same depth away from each eye even though it may be at a slightly different distance from each of the nodal points along each of the viewlines.

We will now consider some geometry involved in Figure 3.28. Firstly, note

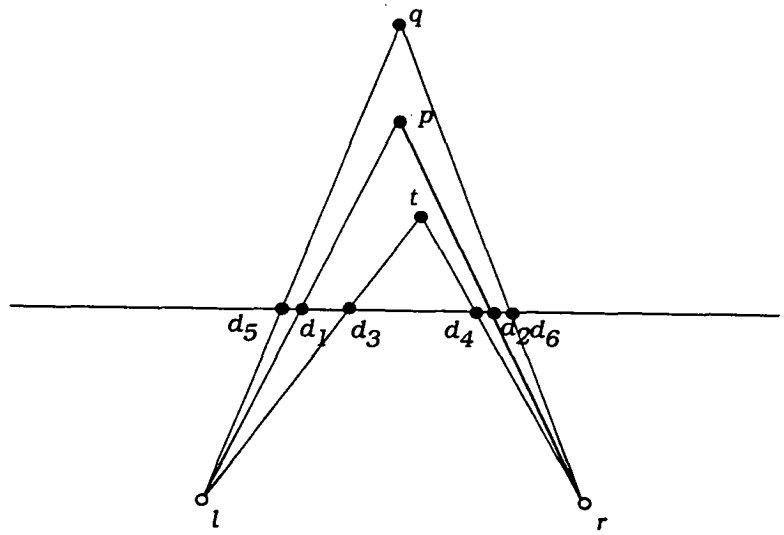


Figure 3.29: Dots d_1 and d_2 , perceived point p ; dots d_3 and d_4 , perceived point t ; dots d_5 and d_6 , perceived point q

that if d_3 and d_4 are matching dots which are closer together than d_1 and d_2 , then from Figure 3.29 we would expect their perceived image, t , to be closer to us than p . Similarly, if d_5 and d_6 are matching dots which are further apart than d_1 and d_2 , then we would expect their image, q , to be further

away from us than p . This is also obvious from the result of equation 3.1.

We will now investigate an extension of our two dot cases in order to check whether this last result fits the practical situation. Consider a horizontal row of equally-spaced dots such as those in Figure 3.30. If we view this row



Figure 3.30: Row of equally-spaced dots

with either of our viewing techniques (described in section 3.2.1), then we do indeed ‘see’ a row of dots which we could describe as being on a line parallel to our original row. For the ‘uncrossed’ viewing technique this row is further from us than our page, and for the ‘crossed’ viewing technique this row is between us and the page (see Figure 3.31). It is to be noted that this is similar to the effect obtained when we view a wallpaper frieze in a similar manner. To achieve the same effect such a frieze must be a horizontally translated motif. Suppose that we now vary the spacing of the dots in our

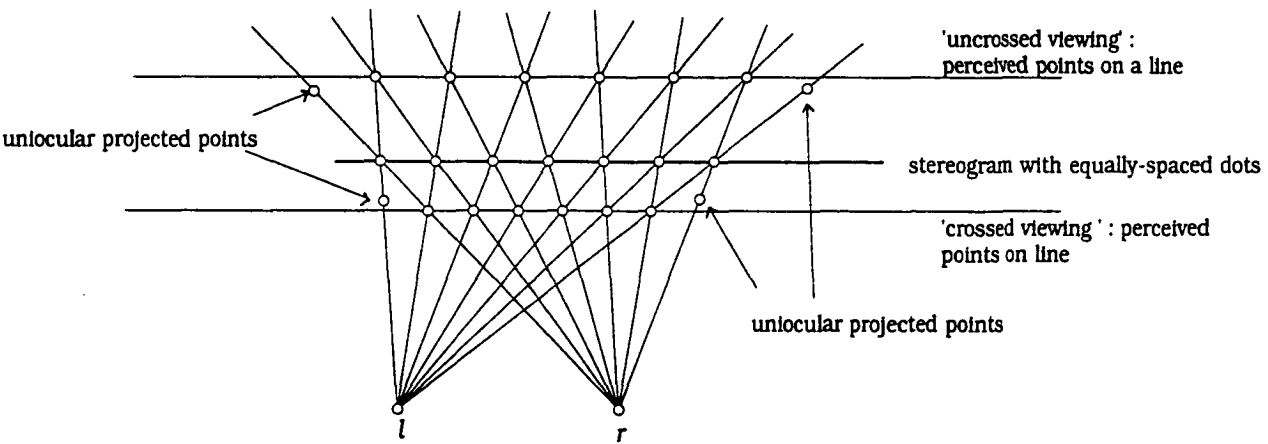


Figure 3.31: Triangulation to give a row of equally-spaced dots

row and view this new row with the same techniques as above. The result



Figure 3.32: Row of unequally-spaced dots

is interesting and perhaps, not what we would expect from the application of equation 3.1. It would appear that the random spacing of identical dots means that it is hard for our brain and eyes to combine to fuse pairs of dots. Does the brain pair consecutive dots or perhaps, every second or third dot? Before attempting to answer this question, we will return to an examination of the row of equally-spaced dots of Figure 3.30. We were easily convinced that we perceive a row of dots on a line, but how did our brain pair, and fuse, the dots to achieve this effect?

The spacing between consecutive dots is the same, but so is the spacing between every second dot (twice that of consecutive dots) and between every third dot etc. By considering the triangulation diagram for each possibility we can obtain some insight (see Figure 3.33). We must keep in mind the fact that both eyes see every dot except, perhaps, for the very edge dots. This mimics the situation on an object such as a cube, where the right eye can see points furthestest to the right which the left eye cannot see, and vice-versa.

Provided that this geometrical model fits the actual situation, we can decide which way our eyes and brain have paired the dots by noting the number of dots which we perceive. For example, if we have n dots and fuse consecutive dots, then the number of perceived dots (including the two dots seen by one eye only) is $n + 1$. If we fuse every i^{th} dot (spacing can't exceed spacing of eyes) then the number of perceived dots is $n + i$. As we could expect from equation 3.1, the bigger the spacing between matching dots, the further away from us is the perceived row (assuming that we don't vary our distance, d , from the row of dots).

At this point we should consider an often-mentioned problem of vision research. This is termed *The False Targets Problem*. A detailed description of this problem can be found in Grimson [8, page 19] and Julesz [13, page 119]. It addresses the fact that although there are many possible ways of

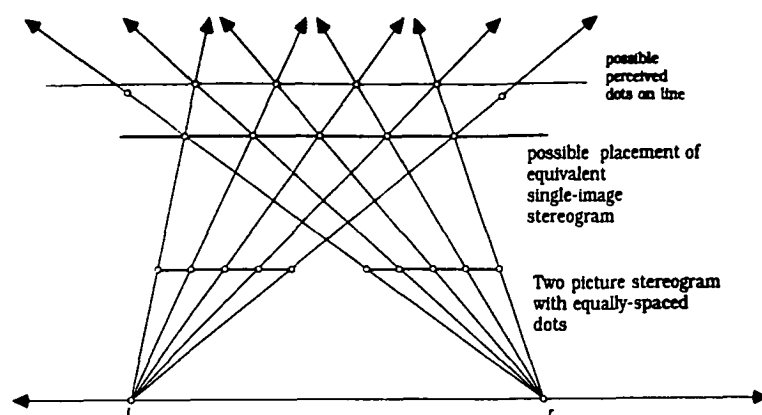


Figure 3.35: Possible perceived dots for two-picture dot stereogram

but the spacing is different from that of the two-picture case. To achieve the same effect as the original case, it must be further from the viewer. This assumes that the geometry represents the true physical and psychological situation.

Consider the following experiment where we again view a row of equally-spaced dots, but this time we will use two guiding dots to aid in our focusing (see Figures 3.36 and 3.37). These guiding dots are to be fused in the same manner as those in Figure 3.26. That is, by focusing behind the page until we have a clear image of three dots as in Figure 3.27. In Figure 3.36, the guiding dots have the same spacing as consecutive dots and in Figure 3.37, they have the spacing between every second dot. In each case, when focusing using the guiding dots, we match the dots of the row which have the same spacing as the guiding dots. This is made clear by counting the number of perceived dots in the row.

Now what happens if the guiding dots are spaced a non-integral multiple of the consecutive dot spacing? In both Figure 3.38 and Figure 3.39, the guiding dot spacing is greater than the consecutive dot spacing (from now denote as s_d).

By fusing the guiding dots of Figure 3.38 and keeping the middle dot as our fixation point, we appear to match every consecutive dot of our row. In

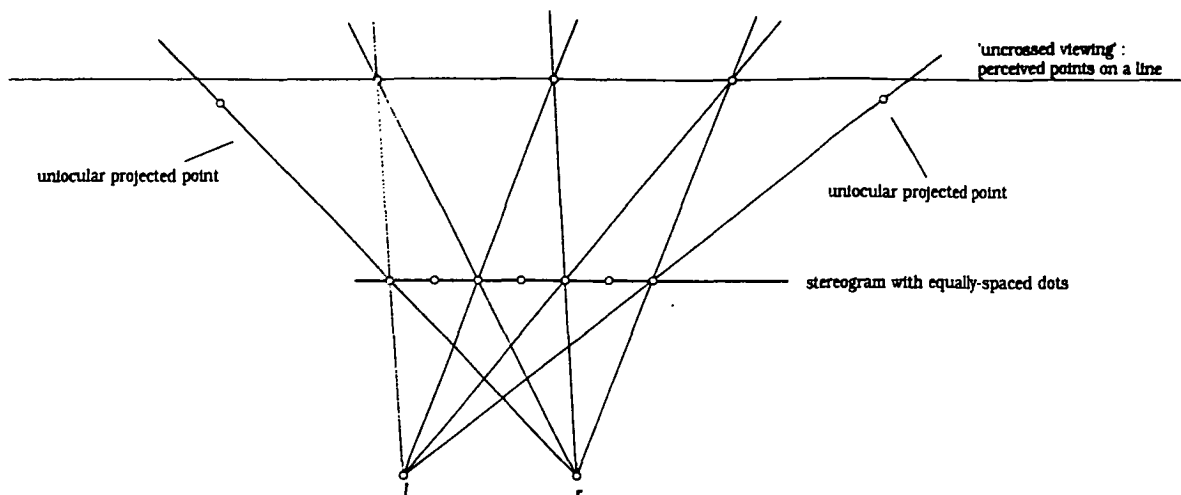


Figure 3.33: Triangulation for every second dot to give a row of equally-spaced dots

matching the dots of Figure 3.34, the brain always selects dots which allow us to perceive dots on a horizontal line. Any other matches may be ruled out as they correspond to physically unlikely situations. For example, there

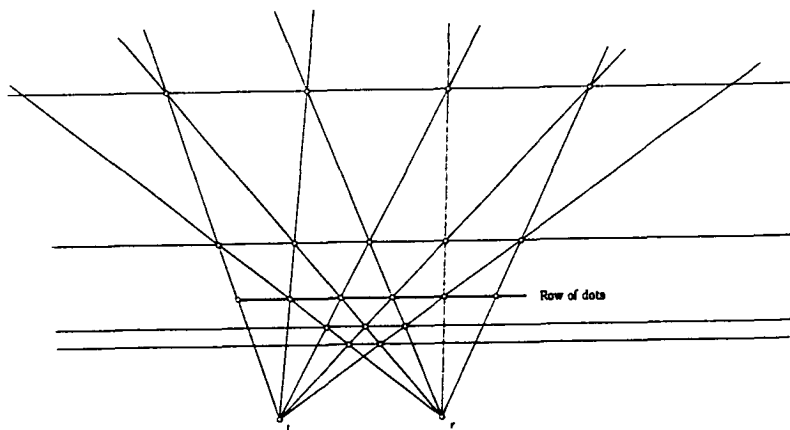


Figure 3.34: Possible perceived dots

should not be more than one match along any line of sight from either eye. It should be noted that if we consider the original false targets problem for two-picture stereograms as shown in Figure 3.35, then it would appear that this two-picture stereogram with equally-spaced matching points can be replaced by a single-image stereogram. This single-image stereogram again has equally-spaced matching points (can be proved using similar triangles)

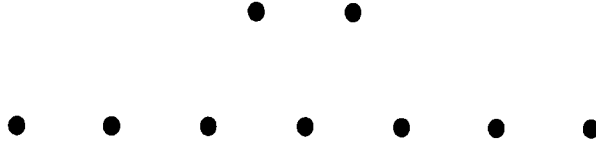


Figure 3.36: Guiding dot spacing is equal to s_d

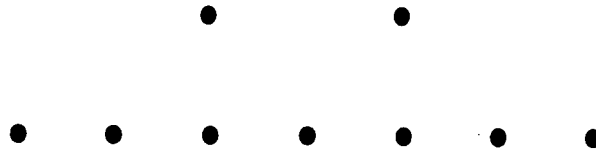


Figure 3.37: Guiding dot spacing is twice s_d

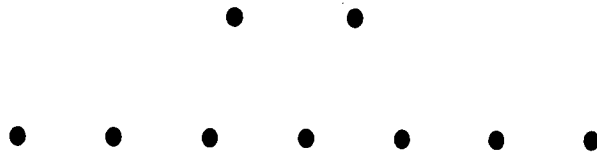


Figure 3.38: Guiding dot spacing closer to s_d than to twice s_d

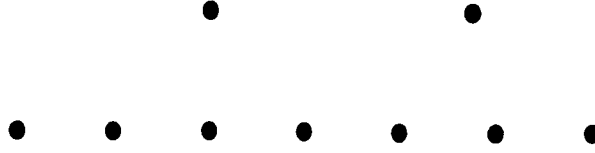


Figure 3.39: Guiding dot spacing closer to thrice s_d than to twice s_d

Figure 3.39, we match every third dot. By measuring the spacing between the guiding dots in each case we could suggest that these results are not surprising. In Figure 3.38, the guiding dot spacing is closer to s_d than to twice that spacing. In Figure 3.39, the guiding dot spacing is closer to three times s_d than to twice that spacing. This could be attributed to the fact that if we are fixated on our perceived image of the guiding dots which corresponds to a particular dot-spacing, say d_f , then our eyes and brain combine to choose sensible pairings. Possibly such pairings have spacing which varies the least from d_f . We will elaborate later in section 3.4 by linking this with our results of section 3.1.

It is appropriate to note the fact that when the guiding-dot spacing equalled the spacing of the matching equally-spaced dots in our rows of Figure 3.36 and Figure 3.37, then the fixated point appeared to be on a plane with the row of dots. That is, it appeared to be the same depth behind the page as the row of dots. We now extend our similar triangle result for one row of dots, which gave equation 3.1, to a result for multiple rows, via the following theorem.

Theorem 3.1 *Suppose we have a plane, Π , a fixed line L not in Π , such that L is parallel to Π , and any two points, $s \in \Pi$ and $u \in L$ (see Figure 3.40). If the ratio, $|\overline{p \vee s}|/|\overline{s \vee u}|$, of the line segments $\overline{p \vee s}$ and $\overline{s \vee u}$, is a constant value, k , where p is a point not on L , and not in Π , then the locus of such points, p , for given fixed k , is a plane, Π' , parallel to Π .*

Proof: Choose $t \in \Pi$, such that $u \vee t \perp \Pi$, and produce $u \vee t$ to a point p' ,

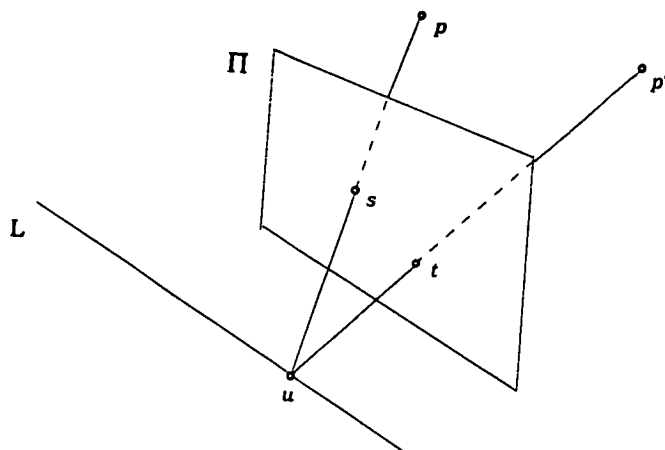


Figure 3.40: Plane Π ; fixed line L ; points $s \in \Pi$ and $u \in L$

where

$$|\overline{p' \vee t}| / |\overline{t \vee u}| = k.$$

Now $\triangle pup'$ is in the plane $p \vee u \vee p'$ and $s \vee t$ divides the sides of $\triangle pup'$ in proportion. That is,

$$|\overline{p \vee s}| / |\overline{s \vee u}| = |\overline{p' \vee t}| / |\overline{t \vee u}| = k \text{ (given)}$$

$$\text{therefore } s \vee t \parallel p \vee p'$$

and so

$$p \vee p' \in \Pi' \text{ where } \Pi' \parallel \Pi.$$

Now $u \vee t \perp \Pi$ (construction)

$$\text{therefore } u \vee t \perp \Pi'$$

and the distance between the planes, Π and Π' , is

$$|\overline{p' \vee t}| = k |\overline{u \vee t}|$$

and this distance is the same for all such points u and t . ◇

Now consider the case of a stereogram with equally-spaced dots. As a consequence of equation 3.1, our constant value, k , of Theorem 3.1, for each row of dots is given by

$$p_d/d = s/e/(1 - s/e)$$

where s is the dot-spacing, e is the viewer's eye-spacing, p_d is the depth of the perceived image behind the stereogram, and d is the distance of the row of dots from the viewer. For each row of dots, we have a different value for each of p_d and d . However, the ratio, p_d/d , is fixed. Consequently, the images of equally-spaced dots lie on a plane which is parallel to the original plane of the stereogram.

We have an analogous result for the case when the viewer uses "crossed-eyes" viewing. In this case, the constant ratio is given by

$$p'_d/d = s/e/(1 + s/e)$$

where $p'_d > 0$ is the distance of the perceived image in front of the stereogram.

Again the geometry fits our practical experience. This is a proof of the so-called *Wallpaper Effect*. That is, if l and r represent the positions of our eyes, then we have the result that:-

Rows of horizontally equally-spaced dots, when viewed appropriately, present a perceived image which lies on a plane parallel to the plane of the original dots. This plane will be behind the stereogram plane in the case when it is viewed with "uncrossed" eyes, and between the stereogram plane and the viewer, when viewed with "crossed" eyes.

Figure 3.41 and Figure 3.42 allow us to check this result. In the first case, the matching-dot spacing is obviously fixed, however this is not so obvious in the second case. In this latter case the brain cleverly pairs the dots with equal spacing. How the brain does this cannot be answered conclusively, however, further investigations provide a possible explanation. We must again note that the end dots on each row of Figure 3.41 which are viewed by one eye only, appear to be closer to us than the perceived images of the paired dots. This is much more noticeable in the multiple row case than it was for a single row.

We will now again investigate the addition of guiding dots to help us focus at an appropriate depth to enable us to fuse the images of the correct matching pairs of dots.

In the case of Figure 3.43, we obtain similar results to those obtained in the single row case. We can force our eyes and brain to match consecutive dots or every second dot, according to our guiding dots. However, in the

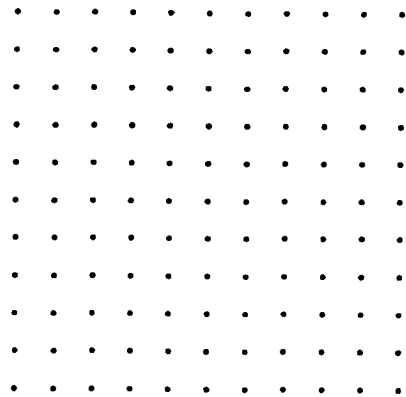


Figure 3.41: Obvious equally-spaced dots

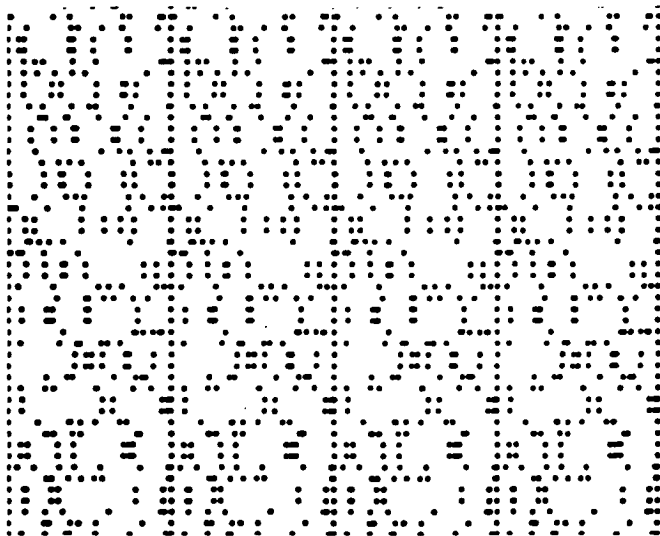


Figure 3.42: Not so obvious equally-spaced dots

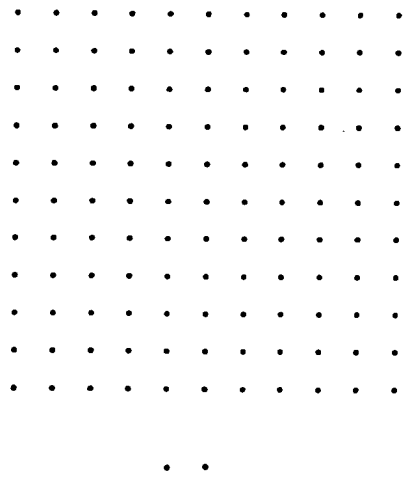


Figure 3.43: Guiding-dot spacing is that of consecutive dot spacing

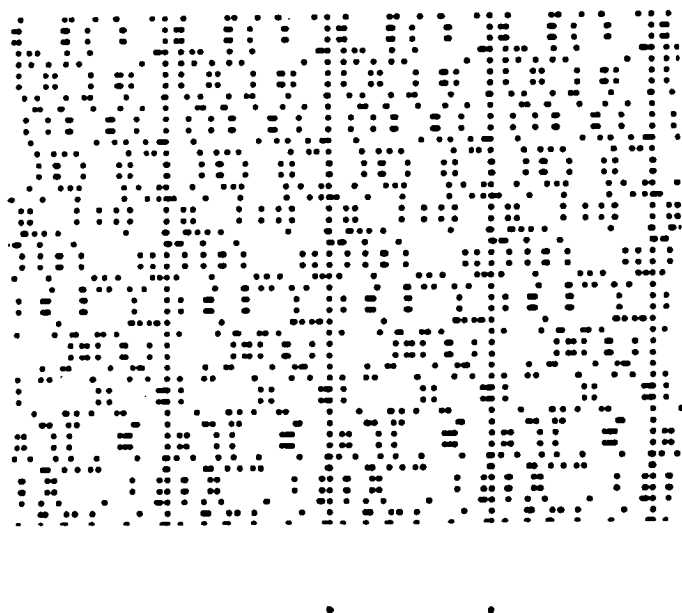


Figure 3.44: Guiding-dot spacing given by horizontal spacing of vertical columns

case of Figure 3.44, we cannot speak of matching consecutive dots or every second dot. In this latter case, the vertical columns of dots give us some inkling as to how the dots should be matched. This spacing seems appropriate for the guiding dots. Once we have focused at the depth defined by these guiding dots, then the brain pairs other dots of all rows with this same spacing. This is evident as the whole picture is perceived on the same plane as the perceived image of the guiding dots. If we use guiding dots which are considerably closer together, as in Figure 3.45, then we find that all we can see is a jumble of dots. These dots do not present a clear image. However,

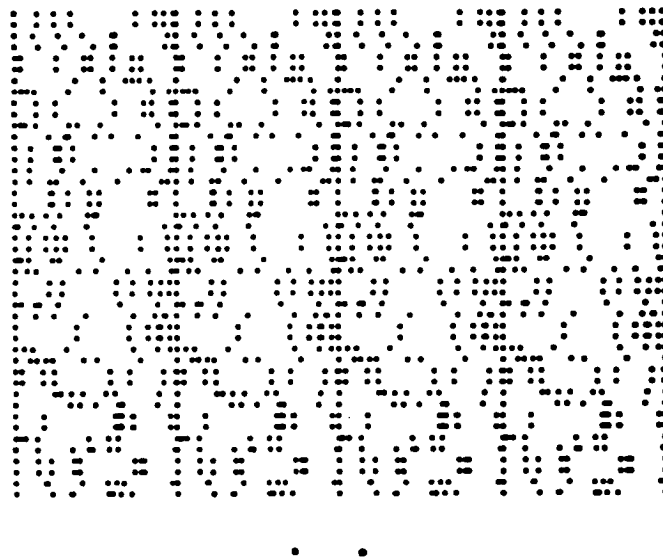


Figure 3.45: Guiding-dot spacing much smaller than that given by horizontal spacing of vertical columns

if we double the distance between the original guiding dots of Figure 3.44, then again we have a planar image which is further away from us. Mathematically, using the result of equation 3.1, if we double the dot spacing, s , then the depth of the perceived dot is more than double the depth of the perceived dot for the original spacing. This is assuming that all other variables remain fixed. It must be noted that we can only judge relative depths of different planes. It is impossible to gauge whether the plane we see is at the exact depth which we would predict from equation 3.1. It appears that the clear vertical columns of dots in Figure 3.44 are analogous to the columns of dots in Figure 3.41. Suppose the horizontal spacing

between the dots of the columns is s' . Additional randomly spaced dots occur in the first left-hand interval of width s' , of each row. Each of these random dots must then be paired with a dot which is a distance s' from it along a row. This next dot is then matched with another dot a similar distance along the row. This process continues across the rows. Diagrammatically, this situation for one row of dots is represented in both Figure 3.46 and Figure 3.47.

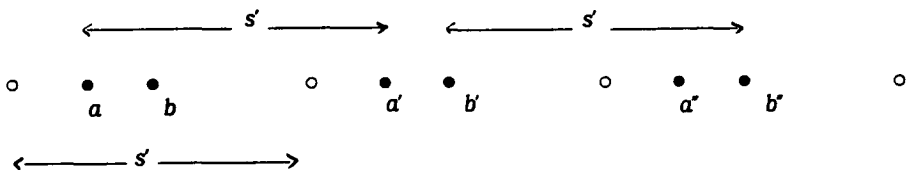


Figure 3.46: o represents dots of vertical columns;• represents random dots

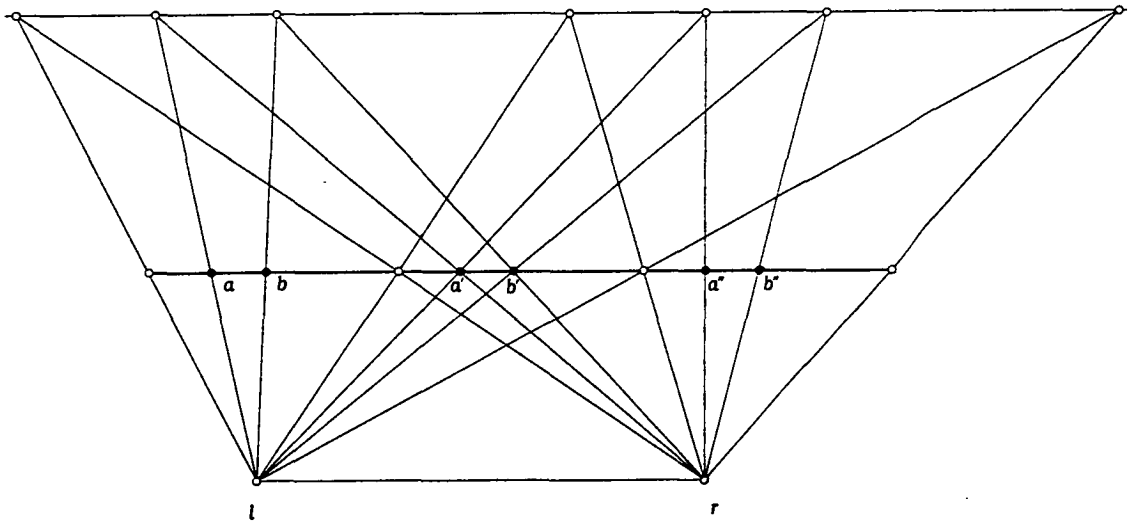


Figure 3.47: Viewlines from each eye meet on a line

Now suppose we remove the vertical column of dots so that we have no obvious clue as to how the dots are to be paired. This time view Figure 3.48

using the uncrossed technique of section 3.2.1. Although it may take a few seconds, our eyes and brain are able to pair the correct dots even though there are no obvious guiding dots.

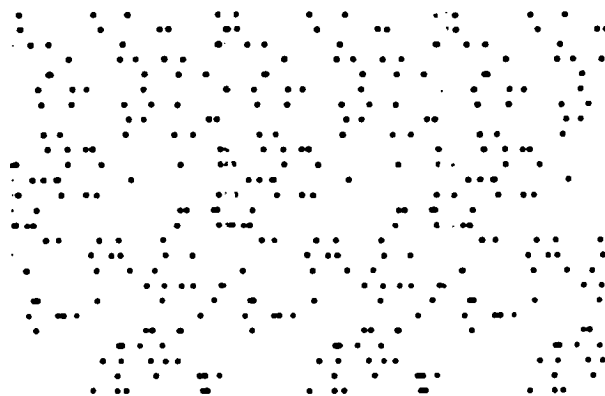


Figure 3.48: No obvious guiding dots

In every situation we have considered so far, our eyes and brain have been matching pairs of equally-spaced dots. Suppose now, that we return to our row of dots and firstly change the spacing between one pair of dots as in Figure 3.49.

As we would expect from the geometry shown in Figure 3.50, we see one



Figure 3.49: Two dots with smaller spacing from the rest

dot closer to us than any others. This is also obvious from the application

of equation 3.1.

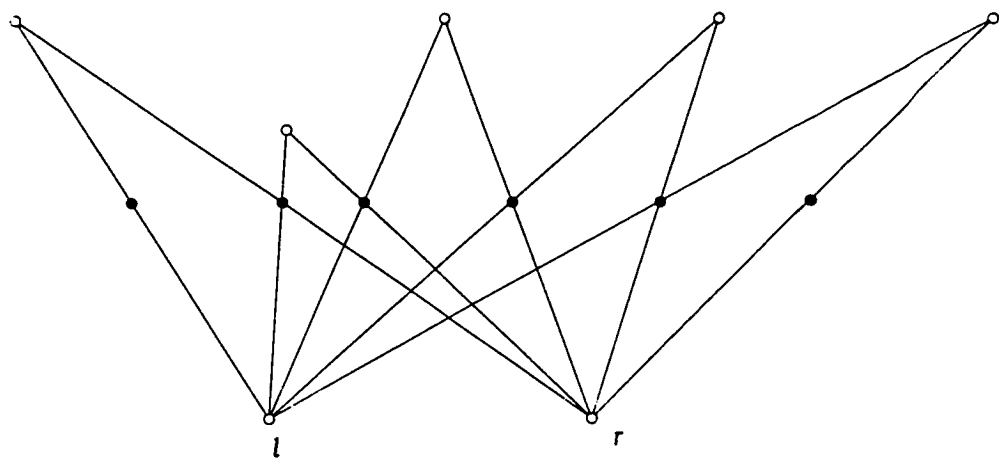


Figure 3.50: Geometry of the preceding case showing viewlines from each eye

Now we will consider a row of dots such as that in Figure 3.51. Here the spacing between consecutive dots is different for every pair. We will view



Figure 3.51: Consecutive dot spacing varies across the row

this row firstly, without guiding dots in Figure 3.51, and then, with two sets of guiding dots as in Figure 3.52 and in Figure 3.53.

By counting the number of perceived dots we can begin to discover which pairs of dots are being matched. Some important points to note are :-

- (i) In contrast to the rows of equally-spaced dots, the perceived images for these two cases (matching consecutive dots and matching every second dot),

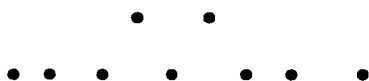


Figure 3.52: Guiding dots 1: consecutive dot spacing varies across the row



Figure 3.53: Guiding dots 2: consecutive dot spacing varies across the row

have different contours. This can be further demonstrated by considering the geometry of Figure 3.54. This gives us the relative placings of the perceived dots when viewed from above.

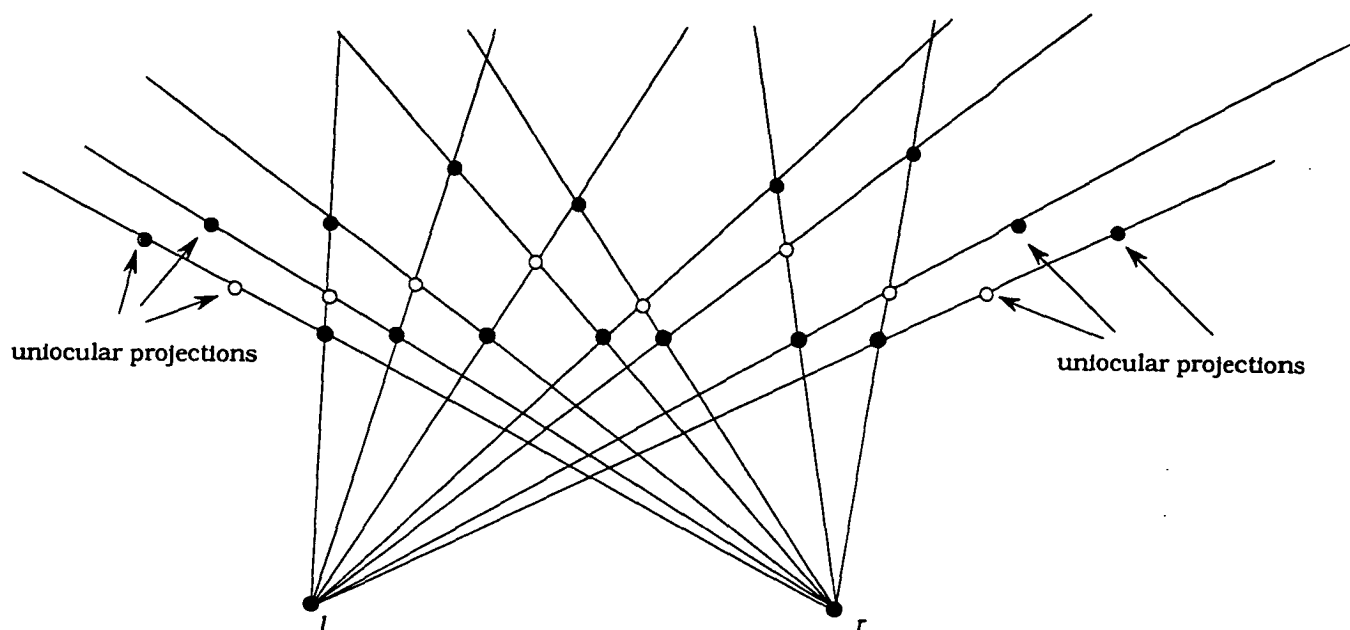


Figure 3.54: ● row of unequally-spaced dots;○ perceived dots: matching consecutive dots;○ perceived dots: matching every second dot

In other words, the pairing, and fusing, of the correct dots is crucial if we are to perceive a given image. This seems to depend on the plane of fixation of our eyes. By *plane of fixation* we mean the perceived plane defined by the dot spacing of our fixation point. Similarly, we define a *line of fixation* which is the line of intersection of this fixation plane with any plane through the viewer's eyes and a row of dots. When we have guiding dots, we have some inkling as to the positioning of the fixation plane. If we do not have guiding dots, then our eyes and brain appear to automatically adjust to an appropriate fixation point, and hence, plane. This will be discussed in more detail after further experimental demonstrations.

(ii) This row of unequally-spaced dots promotes a perceived picture which has clear differences in depth. In our previous example in Figure 3.32, we were unable to fuse the dots to obtain such an image. The explanation of this phenomenon is most likely linked with Panum's Fusion Space which we discussed earlier in this chapter. This possibility will be checked after further discussion.

We now extend our discussions of unequally-spaced dots in a row to unequally-spaced dots in our multiple row cases. Analogous results are obtained. As the dot spacing varies, for matching dots, so does the depth of the perceived point. After fixating behind Figure 3.55 with our ‘uncrossed’ technique, we have four dots on a closer plane and four dots which appear on a most distant plane. The bulk of the dots appear on a central plane. Provided we

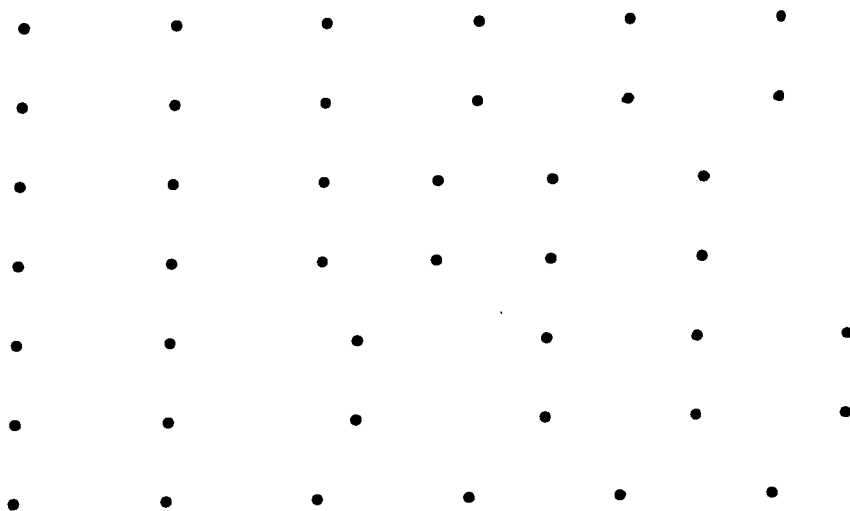


Figure 3.55: Changes in horizontal dot spacing

fuse consecutive dots of every row, this is exactly as we would predict after application of theorem 3.1. If, however, we force ourselves to fixate on a plane defined by the guiding dots given in Figure 3.56, then we perceive a different image.

It is interesting to note that if we don't have guiding dots, we can obtain some rough idea of the fixation plane chosen by our brains. This is with the aid of a clear perspex ruler or a numbered horizontal grid on a strip of overhead projector film. If we place this across our dots of Figure 3.55 (parallel to the rows), after we have a fused image, then we note the superimposition of the numbers on the ruler. For example, if the 3 is superimposed on the 5, we know that we are fixating on a plane in the vicinity of a plane defined by a dot-spacing of two units. *This must be done very speedily before our brains fuse similar markings on the ruler.*

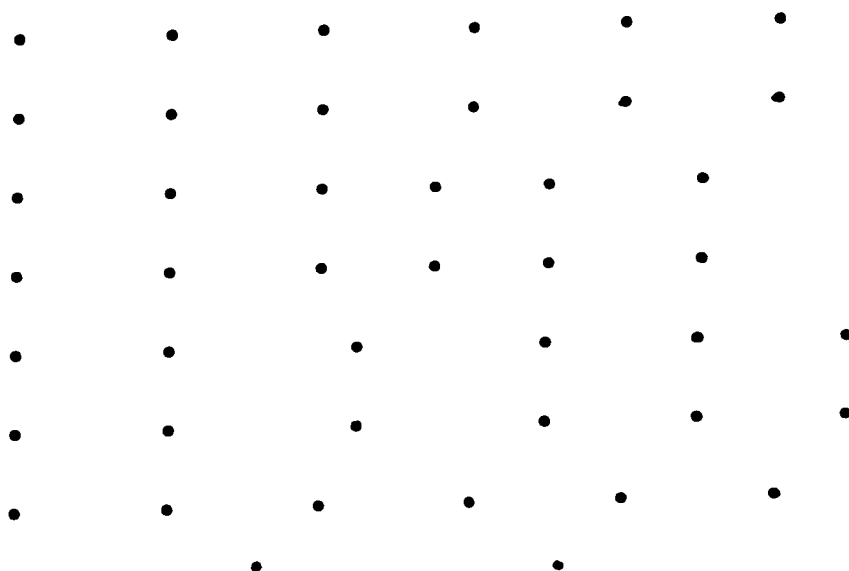
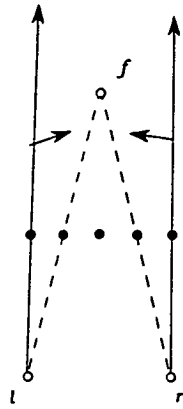


Figure 3.56: Guiding dots force fixation on another plane for fusing dots

Another point to note is that in the cases where more than one picture in depth can be perceived, the viewing technique chosen by the viewer is crucial. Particularly, if there are no guiding dots given. The possible scenarios are illustrated in Figure 3.57. In the case where the viewer begins by staring into the distance, we can imagine that the viewlines are initially almost parallel. Consequently, as the viewlines converge on a closer point to the viewer, they are likely to lock onto a fixation plane with a bigger dot spacing than would be the case if the reverse occurred. That is, if the viewer initially focused on a point on the page of dots and then, gradually diverged the viewlines.

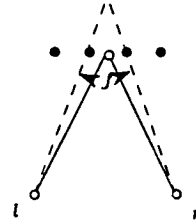
3.3.1 Extension of dots to bigger rectangles

We now consider a combination of the preceding cases. Suppose that we replace our dots of Figure 3.26 with two rectangles such as those in Figure 3.58. The only stipulation being that the distance between matching points on their boundaries, say s , is less than our eye-spacing. By matching points we mean the points such as a and b , that would be superimposed if one rectangle was placed exactly on top of the other. If we fuse these two images



When the viewer begins by staring into the distance, the convergence towards the point, f , could mean the matching of every second dot.

(a)



When the viewer begins by focusing on the page of the dots, the divergence of the viewlines in order to fixate behind the page, could mean the matching of every consecutive dot.

(b)

Figure 3.57: Perceived picture relies on the viewing technique

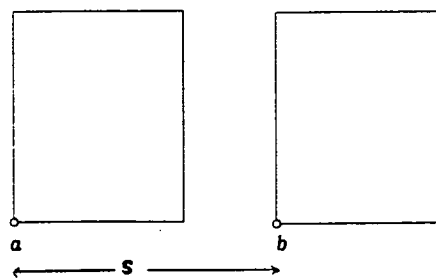


Figure 3.58: Fuse using the 'uncrossed' technique

as described for the dots of Figure 3.26, then we see three rectangles lying on planes parallel to us and our page. It is not clear whether these planes are the same plane. Figure 3.59 shows a geometric view from above. The

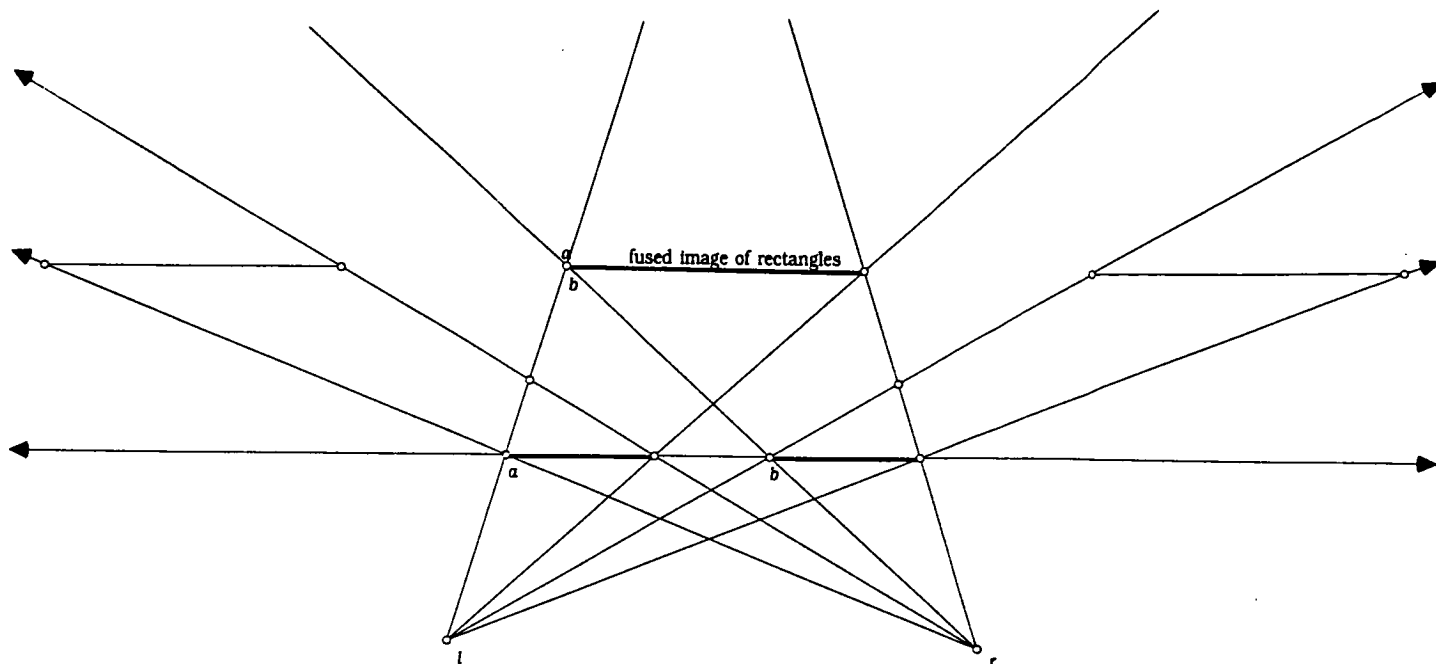


Figure 3.59: Cross-sectional view from above of the fused preceding figure

middle image is the fused binocular image. The left-hand image is a projection of the left-hand rectangle seen by the right eye only. The right-hand image is a projection of the right-hand rectangle as seen by the left eye only. Since these outer projections are of similar size to the fused image, we seem justified in saying that they are further away from us than the plane of the page. Note the superimposition of the letters a and b in the fused central image. Now we add identical circles to the interior of each rectangle so that:

- (i) the distance between the matching points on the circles equals the distance, s , between the matching points of the rectangles. (see figure 3.60).
- (ii) the distance between matching points on the circles is less than s . (see Figure 3.61)
- (iii) the distance between matching points on the circles is greater than s . (see Figure 3.62)

In Figure 3.60, after fusion of the two images, we have an analogous situation to that of Figure 3.58, Figure 3.41 and Figure 3.42. In each case we have a fixed distance between the matching points which must be fused and in each case, the fused image lies on a parallel plane behind our page. In Figure 3.61

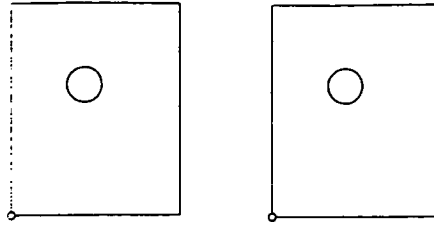


Figure 3.60: Circle spacing = s

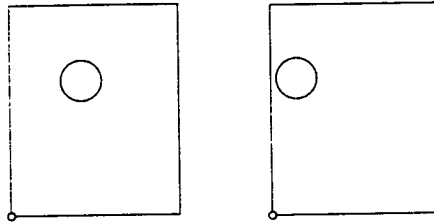


Figure 3.61: Circle spacing < s

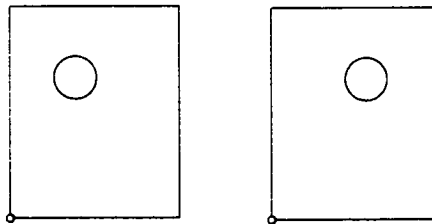


Figure 3.62: Circle spacing > s

the fused middle image gives a circle on a plane nearer to us than the plane of the rectangle. For the outer images which are seen by an individual eye, there is no apparent difference in depth between the circle and the rectangle. In Figure 3.62 the central fused image is a circle which lies on a plane further away from us than the plane of the rectangle.

3.3.2 What can we deduce from these observations?

On fusing the two rectangular images we appear to be fixated on a plane. This plane can be described as being determined by a fixed spacing between matching points of the two planar pictures. In this case, the pictures are the rectangles and the fixed distance is s , the distance between exactly matching boundary points. Adding the circles in the case of Figure 3.60 made no difference to this impression of a planar image, as again, the distance between matching points is s . In fact we could achieve a similar effect by adding a collection of dots to the interiors of these rectangles. The proviso being, that for every dot in the first rectangle, there is a matching dot in the second rectangle a horizontal distance of s from it. This is exactly the situation in the first two vertical strips of Figure 3.42, where we could say that the two rectangles have a common boundary. The vertical boundaries of the rectangles probably behave in a similar manner to our guiding dots of the preceding examples.

It must be noted that the notion of fixating on a plane is not foreign to us. When we normally view a painting, or photograph, we fixate at various points on the plane of the painting via saccadic eye movements which are described in section 3.1. Of course in most instances the artist expects the viewer to stand parallel to it. As we have seen, slant anamorphic art is an exception to this.

Returning to Figure 3.61 and Figure 3.62 we have analogous situations to that of Figure 3.55 where the spacing between some of the matching points does not match the spacing of the supposed fixation plane. When the two pictures are fused, the circles are respectively seen closer to, and further away from, us than this plane.

What we have just described, is analogous to stereoscopic photographs and two picture random dot stereograms such as those of Bela Julesz.

We could ask another question. In fusing the images of the circles with different spacing from s , for example, do our eyes converge more and diverge more to bring the images onto the foveolas of the eyes? That is, do they fixate at alternately different depths?

It seems likely that the answer to this question is NO.

Consider the following experiment. To Figure 3.61 we will add two more identical, and supposedly, matching circles. The horizontal separation of these circles is markedly different from s and they are smaller than the other circles. This is shown in Figure 3.63, where these new circles are labelled 1 and 2. Now if we fuse the two rectangles as before, we find that the

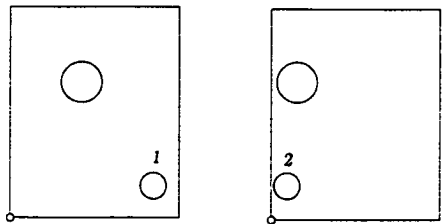


Figure 3.63: two smallest circles with spacing much less than s

bigger circles fuse, but we are unable to fuse the smaller circles. We obtain four images of the smaller circles which appear to lie in a plane close to the plane of the rectangles. Figure 3.64 is a diagrammatical view of the cross-sectional geometry from above. Hence it would appear that the vergence of our eyes is determined by the fusing of the rectangles and larger circles and this vergence does not change to allow the fusion of the smaller circles. If we note the labels 1 and 2 on our small circles when we are viewing the fused image, we find that each circle appears to have been moved horizontally by an approximate distance of s . Figure 3.64 shows the expected position of the image of the smallest circles if they had been fused. Their non-fused images which we see in practice, are the respective projections of their images as seen by each individual eye.

This scenario fits very well with the physiological and psychophysical discussion of stereopsis in section 3.1. That is, when fixating on a point, our

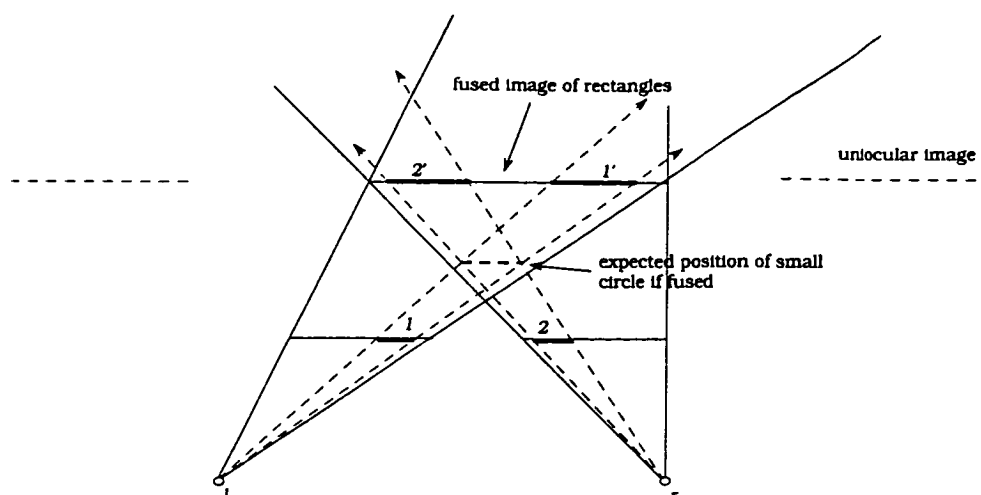


Figure 3.64: Cross-sectional view of geometry of the fused preceding figure from above

eyes and brain can fuse the images of disparate points, within the region of peripheral vision, provided that their images fall on the retina within a small neighbourhood of the foveolas. Such points are said to lie within Panum's fusional space. Images of points which lie outside this region result in double vision or diplopia. This is the situation in the case of the smaller circles of the partially-fused image of Figure 3.64. It was also the case for our nearer finger in the example relating to Figure 3.15 when we fixated on a far finger. In the context of stereograms, what do we mean by disparate points? We consider this question in the next section, but firstly we note the following observation.

3.3.3 Interesting observation

If we fixate on a point behind the page when we are viewing a series of horizontal lines, such as those of Figure 3.65, then we perceive horizontal lines which appear to lie on a plane parallel to, but behind, our page. That is except for two small line segments at each end of the lines. In other words we have an analogous situation to that of the rows of equally-spaced dots. There are various conclusions we could draw:

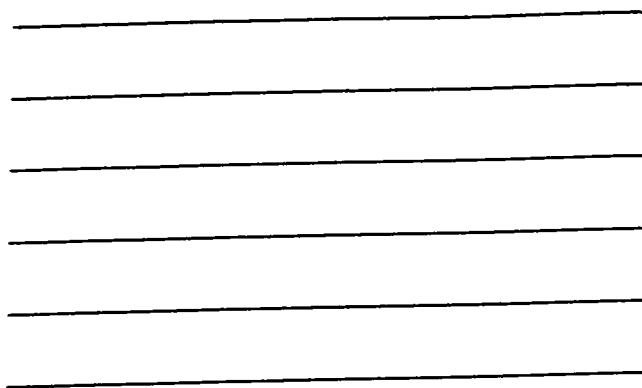


Figure 3.65: Horizontal lines to be viewed using ‘uncrossed’ technique

- (i) Although these lines do not appear to be made up of dots, somehow our brain discerns the miniscule printed dots of the lines.
- (ii) By fixating on a point behind the page, the ‘natural’ result for our brain, given no clues, is to match points which correspond to a dot-spacing on the page of this fixation point.

3.4 Links between the viewing of stereograms and the physiological and psychophysical view of stereopsis

In subsection 3.1.1 we introduced the notion of an horopter, which corresponds to any fixation point. In the context of stereograms, we could say that our fixation plane represents the spacial horopter; or the fixation line represents the horopter on one plane through the eyes. That is, it represents the set of points which the viewer judges to be the same ‘depth’ away from him ...in contrast to the ‘distance’ away from him. It must be noted that this fits Julesz’s [13, page] definition of the horopter. As we have mentioned, ‘distance’ is harder to define. (Although using the results of Luneburg [16], it seems sensible to measure it from the midpoint of the line segment joining the viewer’s eyes.)

This planar horopter corresponds to a particular dot-spacing, say s_f , on a stereogram, for a given position of the viewer (Theorem 3.1). We have seen

that when a viewer is fixating on a point on this horopter, any point which is nearer (in terms of depth) will be represented by dots with a dot-spacing less than s_f and any point which is further away will be represented by dots with a larger dot-spacing. This corresponds to viewlines moving in a similar way to the illustrations for a near point in Figure 3.13 and for a far point as in Figure 3.14. The only difference being that in some near point cases for a stereogram, it is possible for this point to lie outside the Vieth-Muller horopter for the particular fixation point as shown in Figure 3.66. The dia-

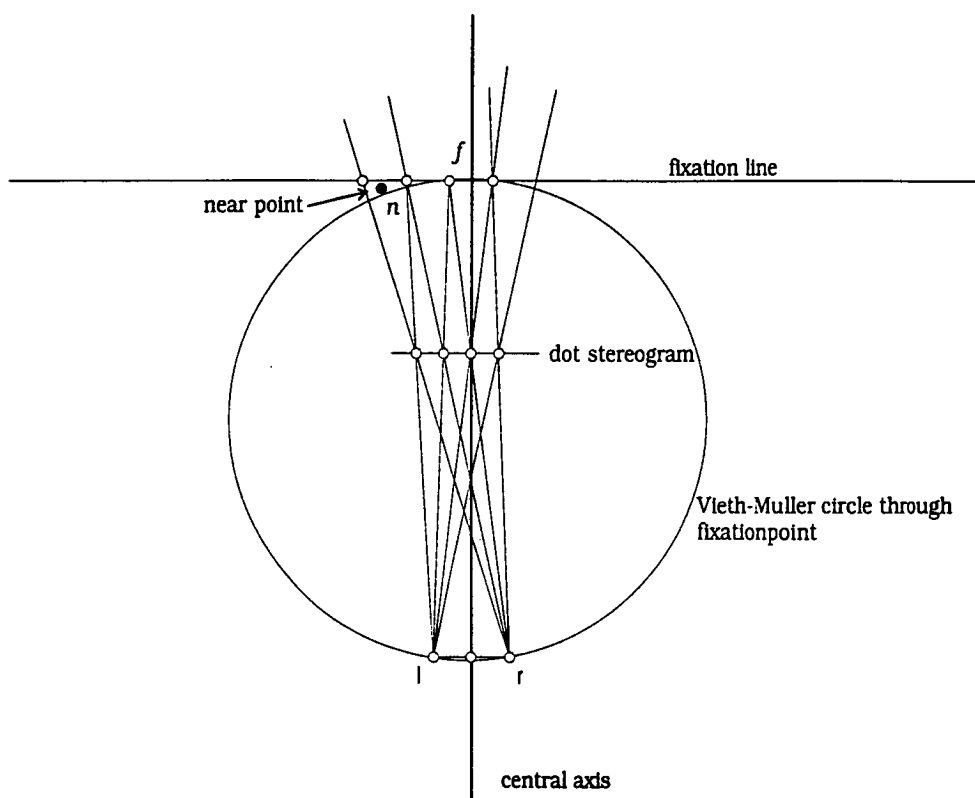


Figure 3.66: Near point, n , for dot stereogram is outside the Vieth-Muller circle

gram in this figure is drawn to scale. It represents the relative placing of a dot stereogram and the Vieth-Muller circle for a viewer observing from 30cm and fixating on a point defined by a certain dot spacing on the stereogram. This viewing distance is appropriate for a viewer of any stereogram presented in this thesis. We can see from this diagram that for normal viewing distances for stereograms, the angle lnr for a near point, n , varies very little from the angle lfr , on the circle. This is provided n is around the *central* perpendicular axis through the midpoint of the segment \overline{lr} . Similarly, the

angle lpr , for any point, p , on the fixation line (again around the central axis), varies very little from the angle lfr . This is relevant when we consider the rough bounds placed upon our binocular fusion by Panum's fusion area. We could now ask the question: How can we describe the notion of disparity in relation to dot stereograms?

It seems that rather than talk about angles as in subsection 3.1.1, we can consider the dot-spacing required to allow us to perceive any point. If we consider the dot-spacing associated with any fixation point f , then we could describe any other point with the same dot-spacing as having *zero disparity*. We have seen that any such point lies on the apparent fronto-parallel plane containing the point f . It is this plane which we have likened to the horopter. *Disparate* points (points with non-zero disparity) will have a dot-spacing which differs from that of the fixation point. In Figure 3.67 (a) we

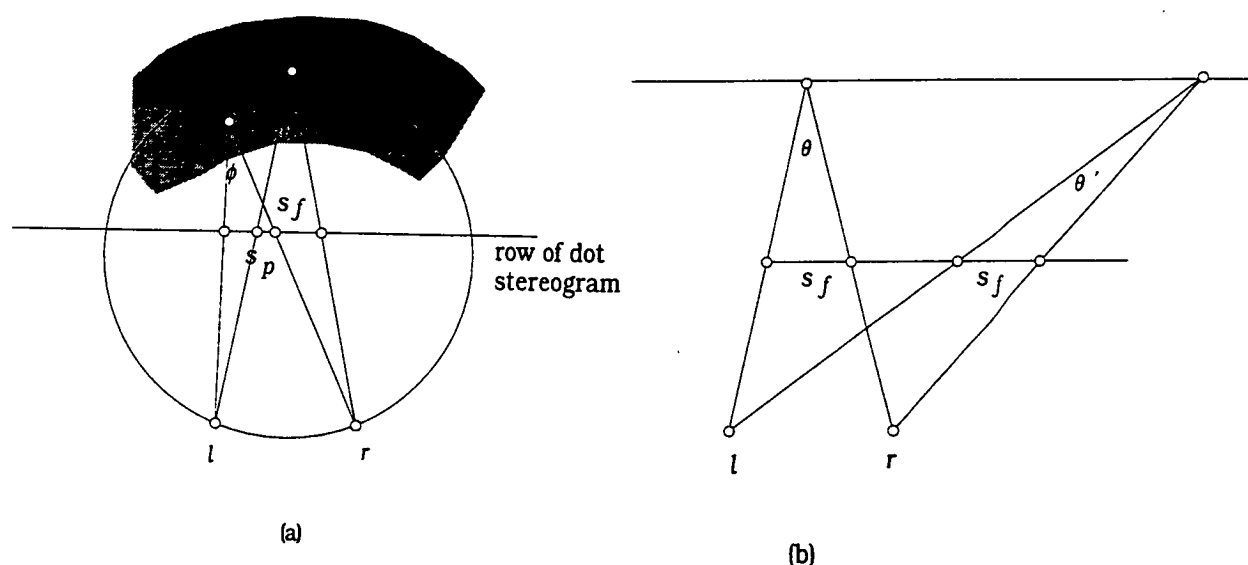


Figure 3.67: (a) Correspondence of angles with dot-spacing; (b) Angle varies slightly for a particular dot-spacing

have re-drawn Figure 3.20 where we have added a representation of a row of a dot stereogram. The dots on this stereogram gives images on the retinae which mimick those of the points, p and f . We see that the angles, θ and ϕ , correspond to the dot spacings s_f and s_p , respectively. Of course a particular dot-spacing does not always give rise to exactly the same angle, for any fixed distance of the viewer as we see in Figure 3.67 (b). However, in the vicinity of the central axis, for a fixed dot-spacing, the angle varies very

little. We are justified in considering this vicinity as Figure 3.4 (b) shows. If we measure two extreme angles such as θ and θ' of Figure 3.67 (b), but on our scale drawing of Figure 3.4 (b), then *Geometer's Sketchpad* tells us that $|\theta - \theta'| \leq 0.56^\circ$. For normal viewing distances of stereograms, the whole stereogram can be seen in our region of peripheral vision and the horizontal width of the stereogram, either side of the central axis, is small relative to the viewer distance.

Now we have seen that for fusion of the images of f and p , we must have $|\theta - \phi| < 2^\circ$. This corresponds to a difference in dot-spacing of $|s_f - s_p|$. One way of placing suitable bounds on this dot-spacing is to think of our stereogram as a screen outside the eyes. We can then make an approximation using Hubel's result of subsection 3.1.3 to translate the 2° difference in angle which corresponds to a distance of about 0.6mm along the retina, to a difference in dot-spacing on our stereogram. Hence if we view a stereogram from a distance of about 30cm, and use Hubel's result shown in Figure 3.21, we find that 0.6mm on the retina corresponds to an approximate distance of 1.05cm along a row. This is provided the viewlines are around the perpendicular from the nodal point to the row of dots. This implies that if the fixation plane of a stereogram corresponds to a dot-spacing, s_f , then in order to fuse the dots of the stereogram which lie within the peripheral vision region, the matching-dot spacing, if it is to be viewed from 30 cm, must lie between $(s_f - 1.05)\text{cm}$ and $(s_f + 1.05)\text{cm}$. Of course the range of values for the appropriate dot-spacing varies according to the distance from the viewer. In designing stereograms (see Chapter 5), this distance from the viewer is crucial, particularly if the stereogram is to correctly represent a mathematical figure such as a cube. The range of suitable values for the dot spacing, corresponds to the perceived picture lying entirely between two planes which are parallel to the stereogram. Maeder [17] may have taken this into account in designing his programs. More details, and a discussion of the desirability of this feature can be found in section 5.1.2.

3.5 Some results of Luneburg

It is interesting to note that Luneburg [16, page 61] makes the point that curves which are apparently straight are of great interest in visual science. He notes that Helmholtz noticed the fact that vertical threads arranged by

an observer in fixed head position to form an apparent fronto-parallel plane are not actually arranged on a physical plane. The shape of the physical surface which appears to be a plane varies with the distance from the viewer. These are described as *horopter curves*. Maybe this is further justification for likening our perceived plane to the horopter. Our discussions relating to the circular horopter in subsection 3.1.1 have so far been limited to a single plane. That is, the plane containing the fixation point and the two nodal points of the eyes. Can this idea of the Vieth-Muller horopter be extended to give us a surface in space?

3.5.1 Experiment 1

We will now present some of Luneburg's suggestions which are based around the idea of the Vieth-Muller circles. Consider his following experiment: construct a number of marks (for example pins) arranged at equal distances on a Vieth-Muller circle through the eyes (see Figure 3.68). If these pins are observed with the head in fixed position, they give the impression of being arranged on a circle with the observer at its centre. He then suggests the possibility, that if we construct part of a torus by rotating the Vieth-Muller circle about the line through the nodal points of the eyes (see Figure 3.69), and then observe this surface again with fixed head, then we may perceive a spherical surface with the observer as centre. This question is left unanswered but he suggests that since different sensations are obtained with a moving head, it could contradict our common experience. Further discussion on this can be found in Luneburg [16, pages 27-28].

3.5.2 Experiment 2

Having experienced the fusing of dots in our preceding section on dot stereograms we are now in a position to try one of Luneburg's experiments for ourselves. In his section on the derivation of the hyperbolic metric of visual sensations he considers the following experiment:

Construct two finite sub-pencils of lines through two points, l and r , which are exactly our eye-spacing apart on a horizontal line. Angles between neighbouring lines of these pencils are the same constant size, see Figure 3.70

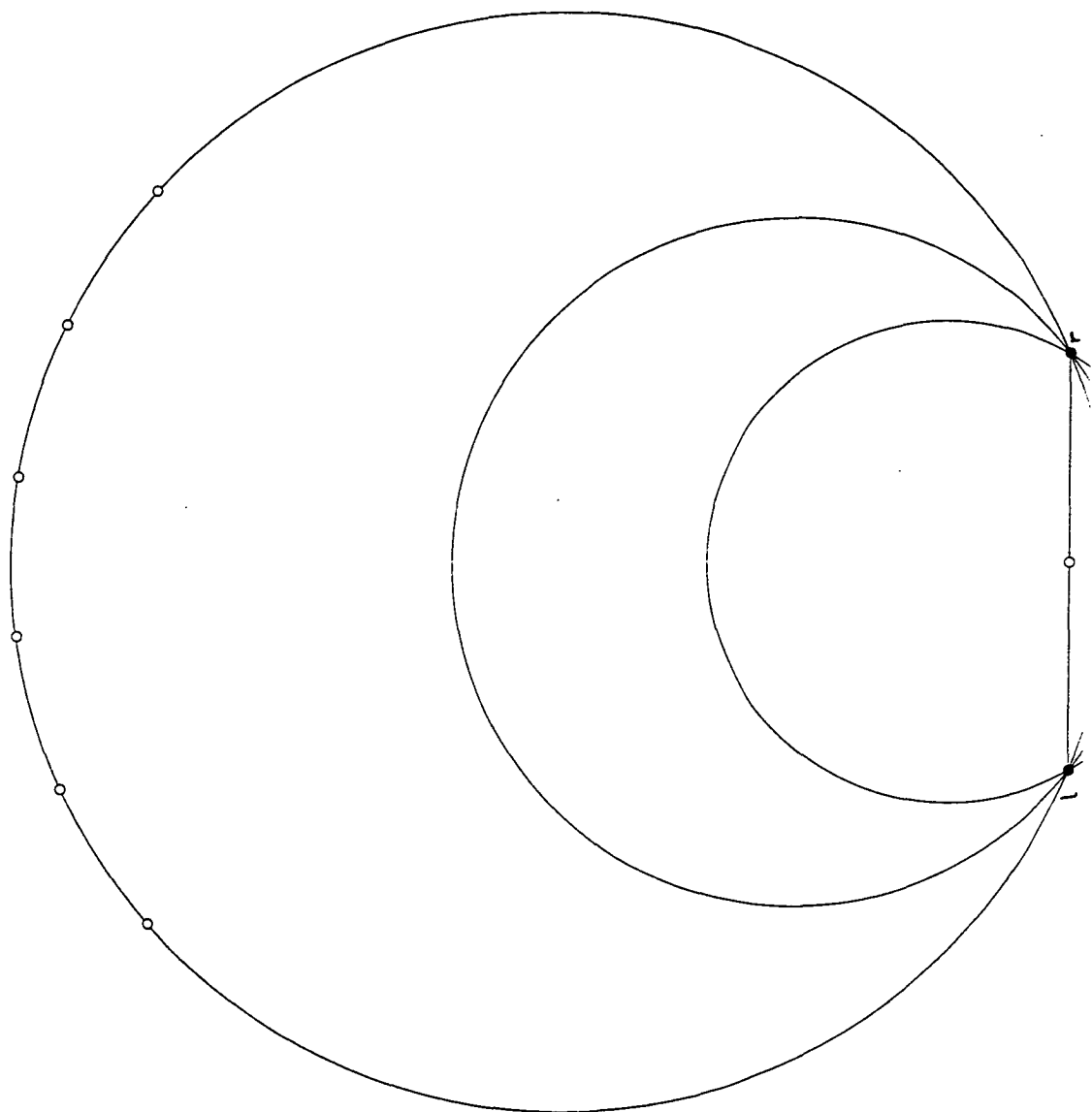


Figure 3.68: Vieth-Muller circle

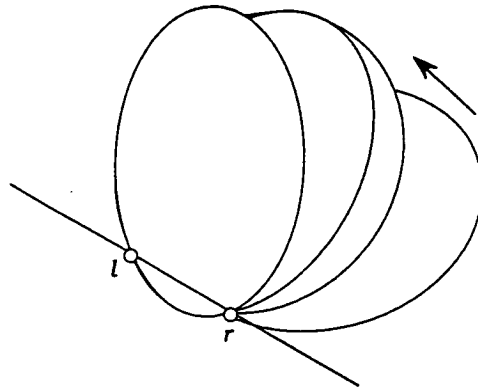


Figure 3.69: Rotation of Vieth-Muller circle about the line $l \vee r$

where, for example, the constant angle is 7.5° . The pencil with the left eye as vertex is red and the sub-pencil with the right eye as vertex is green. The lines of the two sub-pencils intersect one another in pairs on a set of Vieth-Muller circles which determine various points p_0, p_1, p_2, \dots on an axis through the midpoint of the segment $\overline{l \vee r}$ as shown in Figure 3.71.

It must be noted that for the lines to intersect on various Vieth-Muller circles, it is vital that the angle between neighbouring lines is constant. This ensures that we have triangles with the same base-angle sum, which in turn, means that their vertex angles are equal. According to a basic result of Euclidean geometry which says that '*if a straight line joining two points subtends equal angles at two other points on the same side of it, then the four points lie on a circle*', these vertices must then lie on a circle containing the endpoints of the segment, $\overline{l \vee r}$. This is an example of a particular projective correspondence between two sub-pencils of lines. We will consider such correspondences, but infinite, in Chapter 6, where we will see that this Vieth-Muller circle is an example of a conic.

We position our eyes exactly above the points, l and r , of Figure 3.72, and observe the pencils of lines from this position by fixating on some point of the Vieth-Muller circle through p_0 . To make the fusing easier we have eliminated the axis through the midpoint of the eyes in this figure. *Note: For non-practised viewers, it may be essential to view these pencils using the anaglyph 'spectacles'.*

According to Luneburg, instead of two pencils we see one set of fused lines which intersects the Vieth-Muller circle through p_0 at regular distances. He

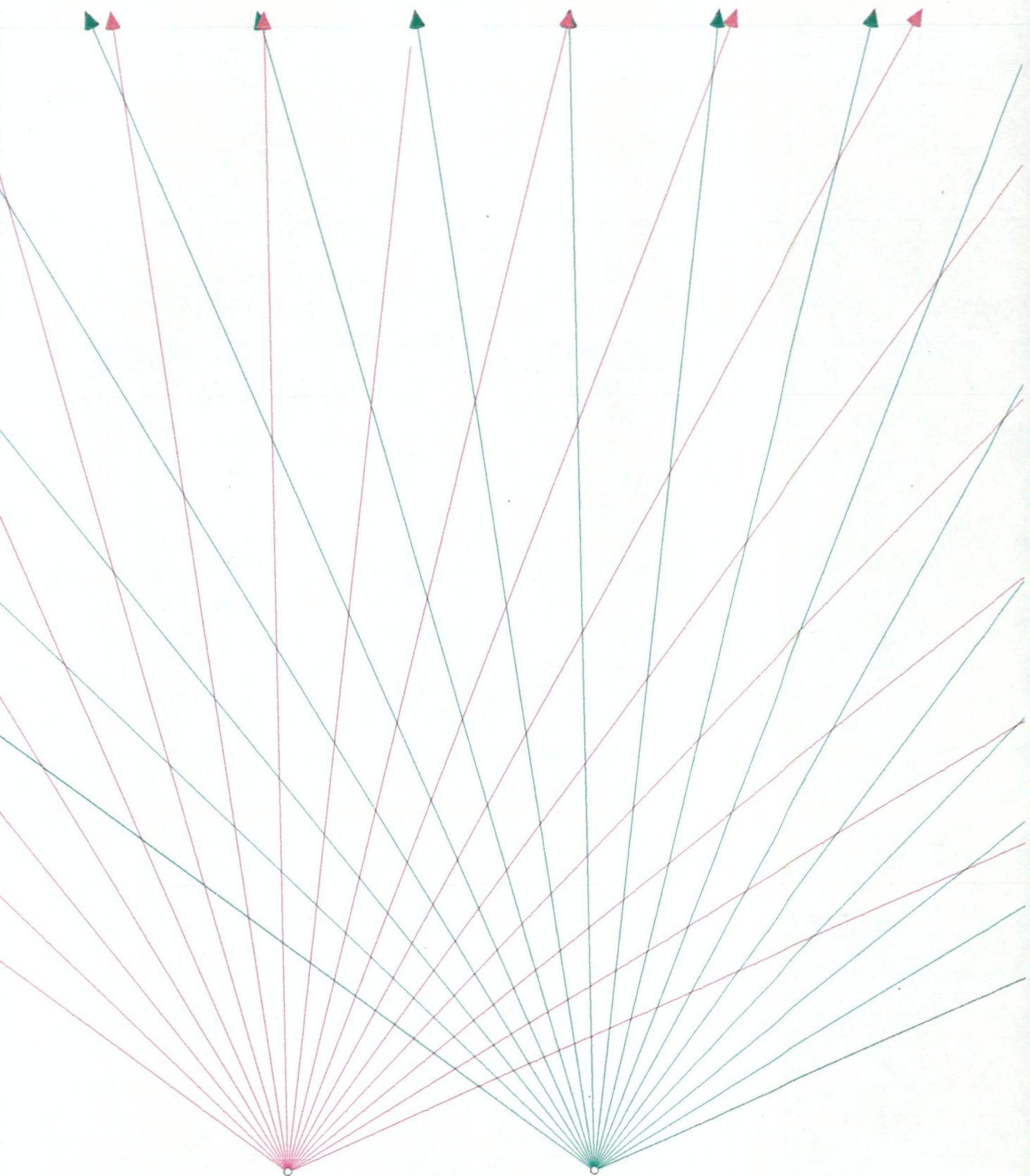


Figure 3.70: pencils of red and green rays \vec{l} with the eyes as vertices

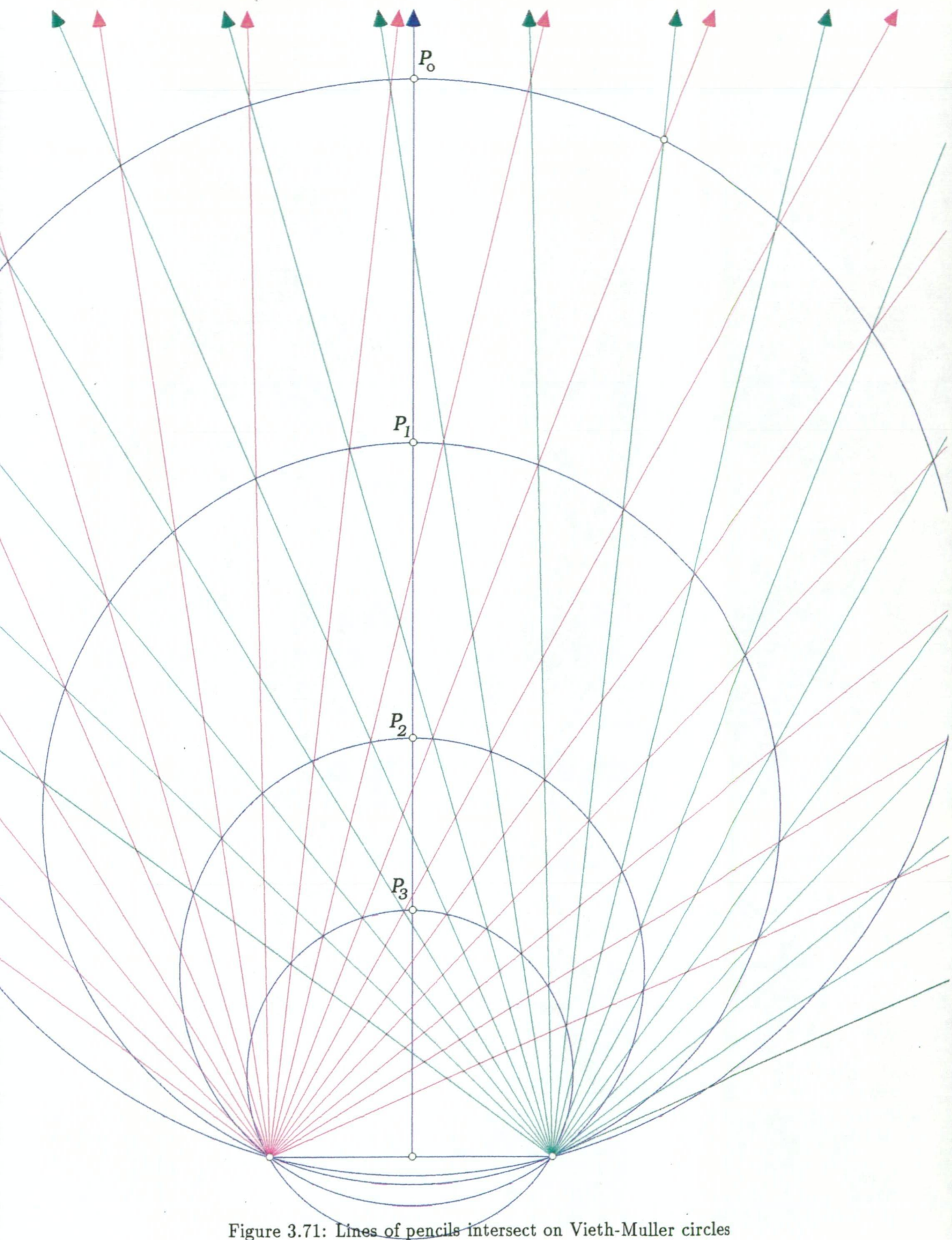


Figure 3.71: Lines of pencils intersect on Vieth-Muller circles

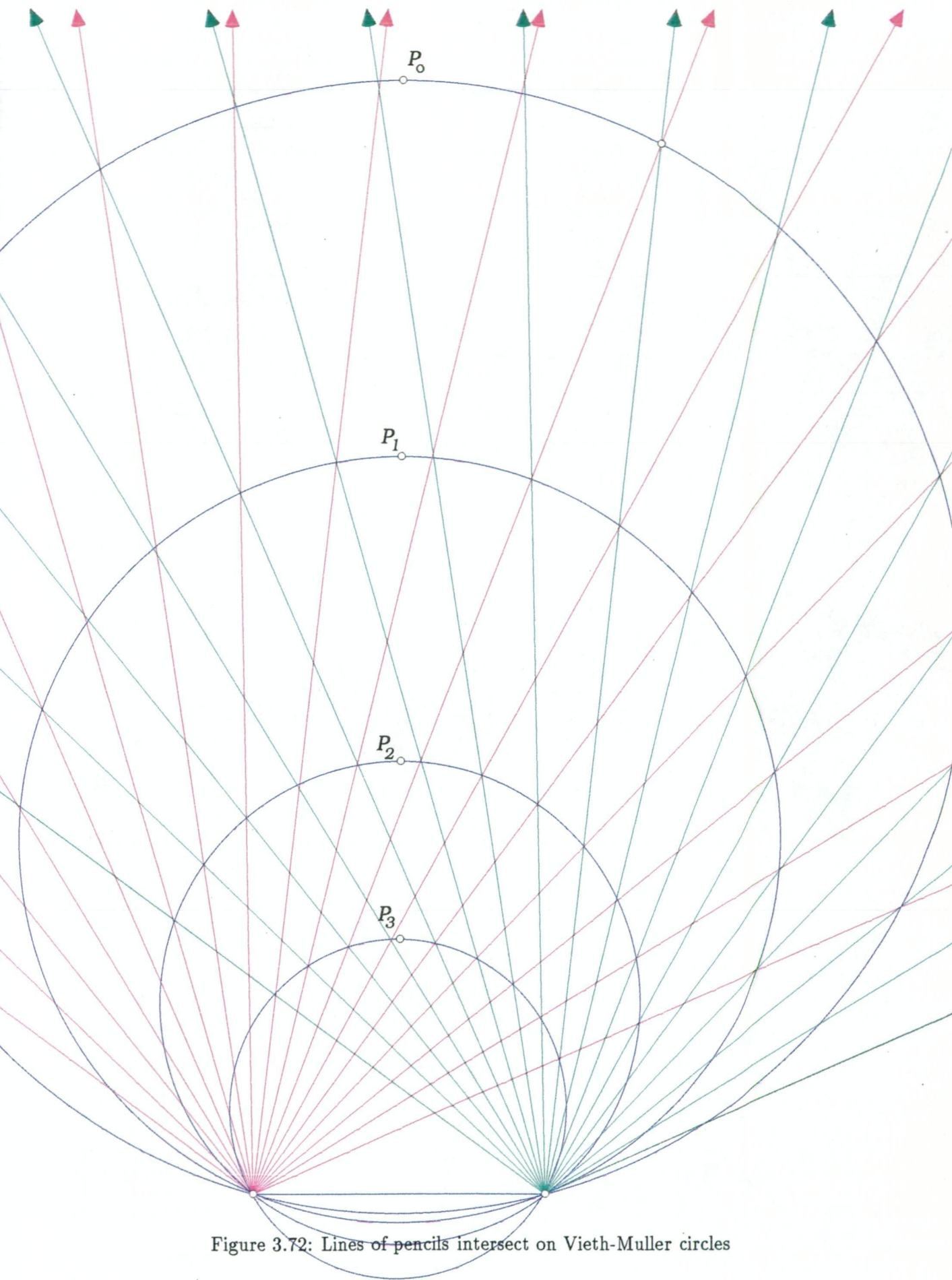


Figure 3.72: Lines of pencils intersect on Vieth-Muller circles

says, “ the appearance of the fused lines is that of straight lines arranged on a circle around the observer.”

Whilst this statement seems correct, we can strengthen it: we see parallel lines which appear to be perpendicular to our page. They intersect the Vieth-Muller circle at the points of intersection of corresponding green and red rays. By corresponding rays, we mean the pairs of rays which our eyes and brain fuse to give the line through the point of intersection. This can be demonstrated by putting distinguishing marks on each of the rays. After fusing, these marks can both be seen on their fused ray.

If we now fixate, in turn, on the closer Vieth-Muller circles through p_1 , p_2 and p_3 , we observe a similar phenomenon in each case. The fused lines again appear parallel, perpendicular to our page and intersect the page at the intersection points of corresponding rays on the relevant Vieth-Muller circle. As the radius of the Vieth-Muller circle decreases, the distance between the fused parallel lines decreases. Using the basic result of Euclidean geometry which says that *‘an angle at the centre of a circle has double the size of an angle standing on the same arc’*, we can show that the intersection points of the corresponding rays on the Vieth-Muller circle are equally-spaced around the circle (in terms of arclength). Hence it is not surprising that the fused lines intersect this circle at, what appear to be, equidistant points. It must also be stressed that these lines ‘look’ parallel. Some justification for this is given in subsection 4.6.2.

Luneburg makes the important distinction between the geometry of the actual physical situation, and the geometry representing the visual sensation. He makes the point that the “Vieth-Muller circle itself appears to the observer as a circle around the point.. of the horizontal plane vertically below the apparent centre of observation.” This statement is confusing. Presumably, by apparent centre of observation, he means the midpoint of the line segment joining the observer’s eyes. Most research, and even Luneburg’s own experiment of Figure 3.68, suggests that such an observation is made if the eyes of the observer are in the plane of the Vieth-Muller circle, not above it, as this statement suggests.

Luneburg suggests that the perceived fused lines can be interpreted as a Euclidean circular cone. He also suggests that a viewer projects any im-

age onto this artificially created cone. Further discussion may be found in Luneburg [16, page 53]. Whilst we cannot dispute his hypothesis, we shall see in subsection 4.6.2 that a cylinder (instead of a cone) of horizontal cross-section equal to the Vieth-Muller circle, and through this circle, seems to provide a good representation of what we see. We will use a computer to demonstrate this possibility.

Chapter 4

Creating Anamorphograms

4.1 Anamorphograms

According to the rule of perspective any picture must be viewed from a particular viewpoint if the viewer is to see what was intended by the artist.

The *anamorphogram* is an extreme example of the special kind of relationship between the viewer, viewpoint and picture. It is a planar picture which deceives the viewer by initially presenting a very distorted image if viewed from the expected viewing position. That is, directly in front at right angles to the plane, such as the viewpoint of you, the reader. As the name suggests—from the Greek *ana* meaning again and *morphe* meaning shape — these distorted pictures must be reformed, or re-shaped, by the viewer. This is achieved by employing unusual viewing techniques from a specified viewpoint, or viewpoints (in the case of both eyes).

Our possible techniques are:

- (i) viewing at a sharp angle from near the side of the picture plane,
- (ii) viewing as a reflection in a suitably placed mirror (planar, cylindrical, spherical or conical),
- (iii) viewing after ‘wrapping the picture around’ a suitably shaped surface
or
- (iv) viewing by focusing both eyes on a suitable plane behind the picture.

It must be noted that in cases (i), (ii), and (iii), viewing with both eyes is not essential.

4.2 Perspective Drawing

We will now mathematically describe the method demonstrated by Dürer and his contemporaries for creating a perspective drawing of an object. Although our method applies to any object for which we can find a representative equation, we will demonstrate this method, and all subsequent methods, by considering their effects on both a rectangular grid of squares (see Figure 4.4) which is two-dimensional and the outline of the edges, or skeleton, of a cube, which of course, is three-dimensional. A three-dimensional image of such a cube may be seen by viewing the anaglyph of Figure 4.32. Creating a perspective drawing of such a cube is often referred to as representing the projection of a wire cube on a screen.

In order to describe our perspective drawings we need the following three definitions. A *figure* is a set, F , of points together with their relationships of collinearity and coplanarity. The *image* of any point, p , from a viewpoint, $v \neq p$, onto a picture plane, P , is the point $p' = (v \vee p) \wedge P$ and is illustrated in Figure 4.1. A *perspective drawing* of any figure, or subject, is the figure whose points are the images of the points of the subject.

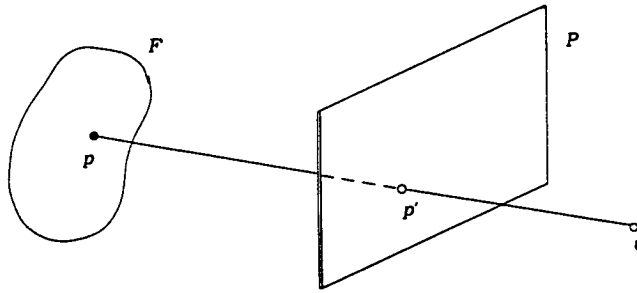


Figure 4.1: p' is image of p ; picture plane P ; viewpoint v

To create a perspective drawing on a page of any figure, F , we will represent the figure by a collection of points in our chosen rectangular cartesian coordinate system of Figure 4.2. We will specify a viewpoint, v , and the equation of the plane containing our page. At this stage we are considering a viewpoint for one eye only (*monocular vision*). For example, in Figure 4.2, for any point, $p = (x, y, z)$, of our figure, F , we need to find an image point, $p' = (x', y', z')$, on the plane, P , given by $y = 0$. It must be noted that, figure

F , could also lie between the viewer and the picture plane. Our perspective

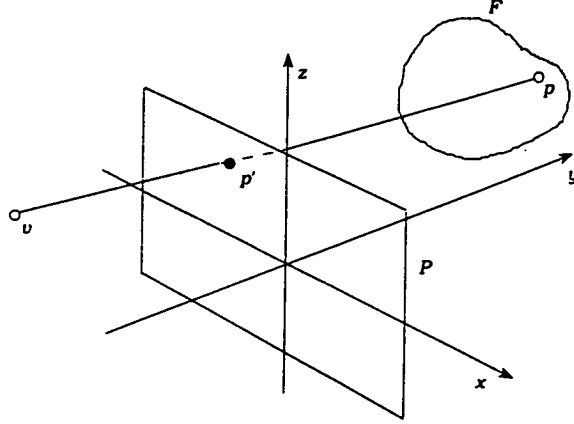


Figure 4.2: Figure F ; p' is image of p ; picture plane P : $y = 0$; viewpoint v

drawing rule of Figure 4.1 indicates that we must find $p' = (v \vee p) \wedge P$, the point where the *viewline*, $v \vee p$, to any point, p of F , intersects the picture plane. The parametric representation of any point on the viewline from the eye, at v , to any point p of the figure is given by

$$v + t[p - v]$$

where $0 \leq t \leq 1$ and the corresponding point, p' , on the picture or drawing, is found by determining the value of the parameter t for which the viewline intersects the picture plane $y = 0$.

We now consider a parametric representation of a rectangular grid.

4.2.1 Parametric representation of a rectangular grid

Suppose this grid lies in the xy plane. If its central point is (x_{centre}, y_{centre}) , we construct suitable horizontal and vertical, intersecting line segments about this point.

Each horizontal line segment will be composed of $2n_x$ smaller segments of length d_x . There will be n_x segments on each side of the vertical line, $x = x_{centre}$, through the centre. Each vertical line segment will be composed of $2n_y$ smaller segments of length d_y . There will be n_y such segments

on each side of the horizontal line, $y = y_{centre}$, through the centre.

The horizontal line segments are represented parametrically by the set of points,

$$(x_{centre} + n_x d_x t, y_{centre} + i d_y)$$

where $i \in \mathbf{Z}$, $-n_y \leq i \leq n_y$ and $t \in \mathbf{R}$, $-1 \leq t \leq 1$ and the vertical line segments are represented by the set of points

$$(x_{centre} + i d_x, y_{centre} + n_y d_y t)$$

where $i \in \mathbf{Z}$, $-n_x \leq i \leq n_x$ and $t \in \mathbf{R}$, $-1 \leq t \leq 1$.

An illustration of our method is given in Figure 4.3, where $n_x = 4$ and $n_y = 2$.

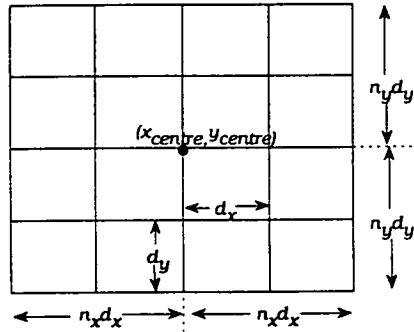


Figure 4.3: Rectangular grid

4.2.2 Perspective drawing of a rectangular grid

One two-dimensional rectangular grid of squares is shown in Figure 4.4, where $(x_{centre}, y_{centre}) = (0, 5)$, $n_x = 5$, $n_y = 5$, $d_x = 1$ cm and $d_y = 1$ cm. Two examples of perspective drawings of this grid are given in Figures 4.5 and 4.6. In each case the viewpoint is $(0, -10, 25)$. In the first case (Figure 4.5), our *object figure*, the rectangular grid, is a set of points on the plane, $y = 0$, and our perspective drawing (referred to as *image figure*) is on the picture plane, $z = 0$. The view from the side is represented in Figure 4.7 (a).

We have what could be described as the 'reverse situation' in Figures 4.6 and 4.7 (b). Here the rectangular grid is on the plane, $z = 0$, and the image

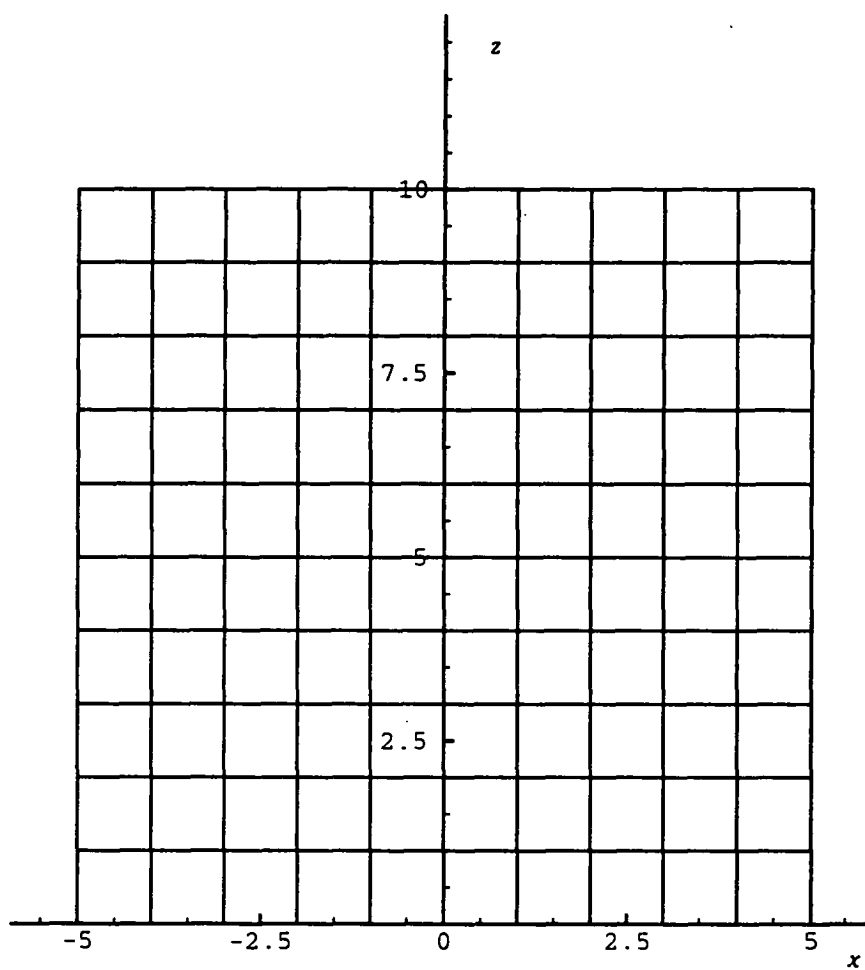


Figure 4.4: Rectangular grid of squares

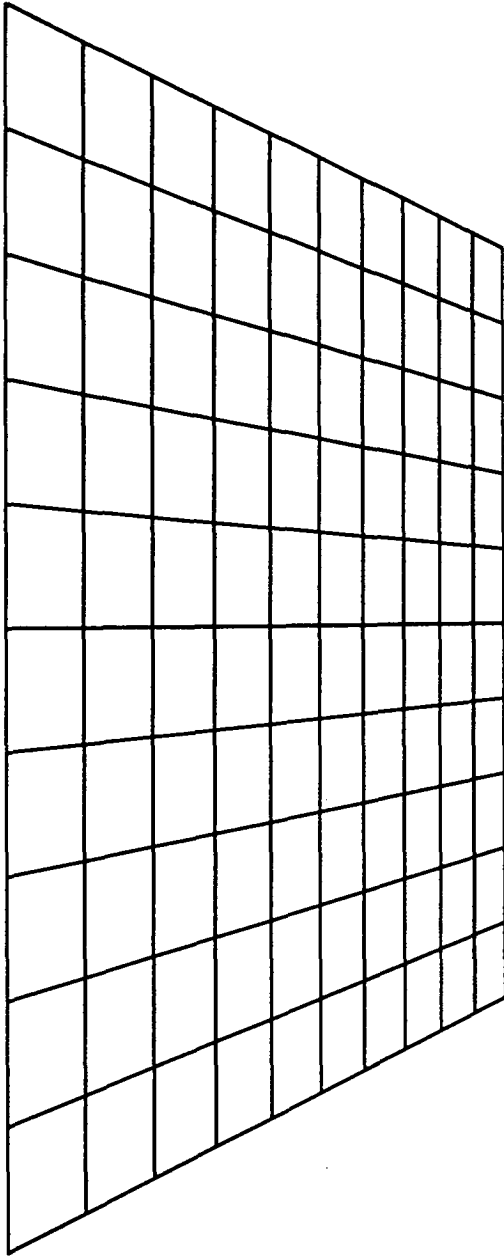


Figure 4.5: Perspective drawing on the plane, $z = 0$, of rectangular grid

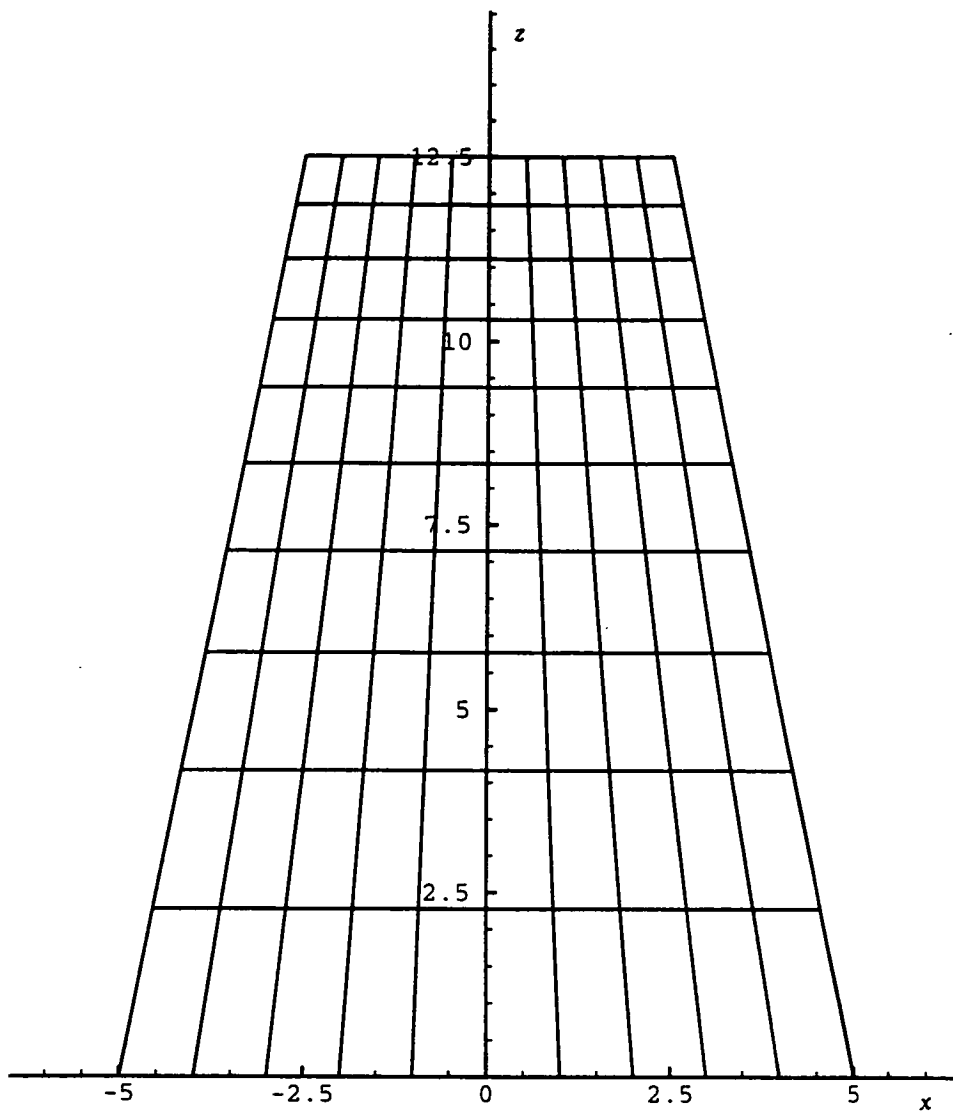


Figure 4.6: Perspective drawing on the plane, $y = 0$, of rectangular grid

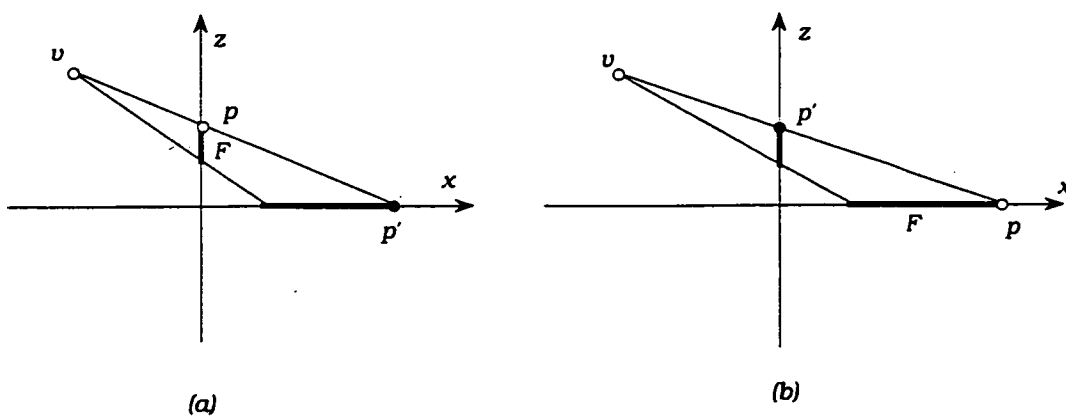


Figure 4.7: Side views showing some viewlines from v to points, p , on our object figure

figure is on the the new picture plane, $y = 0$. In general, it is this situation which an artist mimicks when painting an object or scene which is behind the picture plane. For example, parallel railway tracks moving away from the artist.

Figure 4.7 (a) represents the drawing of what we have referred to as a slant anamorphogram. That is, if the image figure is viewed from v , then this can be described as viewing from a sharp angle from the side. The reader may try this by viewing Figure 4.5 with one eye from an estimated viewpoint of $(0, -40, 20)$ from the point, o , on the horizontal page. The units are centimetres and the page lies in the xy -plane. If Figure 4.7 (a) represents the situation as we see it, then we would expect to see a rectangular grid of squares which appears to be perpendicular to our page, and above our page (possibly the '*floating above the page effect*' mentioned in Leeman, Elffers and Schuyt [15]). Further consideration of slant anamorphograms will take place in subsection 4.3. It must also be noted that for the case depicted in Figure 4.7 (b), we must change our viewpoint from v , if we are to perceive the grid as we see it in Figure 4.4.

We will now consider an appropriate representation of a cube.

4.2.3 Parametric representation of a cube

We need to specify the co-ordinates of the vertices of the cube which, in contrast to our grid, is three dimensional. The program which allows us to create our perspective drawing is included in Appendix B. It allows us to find the vertices of any cube, by beginning with a cube which is centered at the origin as shown in Figure 4.8, where dim denotes half the sidelength of the cube. It can be rotated about any axes, in any order, and then translated in the x, y or z directions. These linear transformations are executed by multiplying each vertex of the original cube by the appropriate transformation matrix using *homogeneous co-ordinates*. For computing purposes, these co-ordinates have the advantage of enabling the representation of all possible linear transformations, by matrices of similar form. Using our normal co-ordinate system, translations cannot be represented using a 3×3 matrix. An added advantage, is that compositions of these transformations to give a net tranformation matrix, allow us to transform about points other than the origin. Some common examples of such matrices are following, and finer

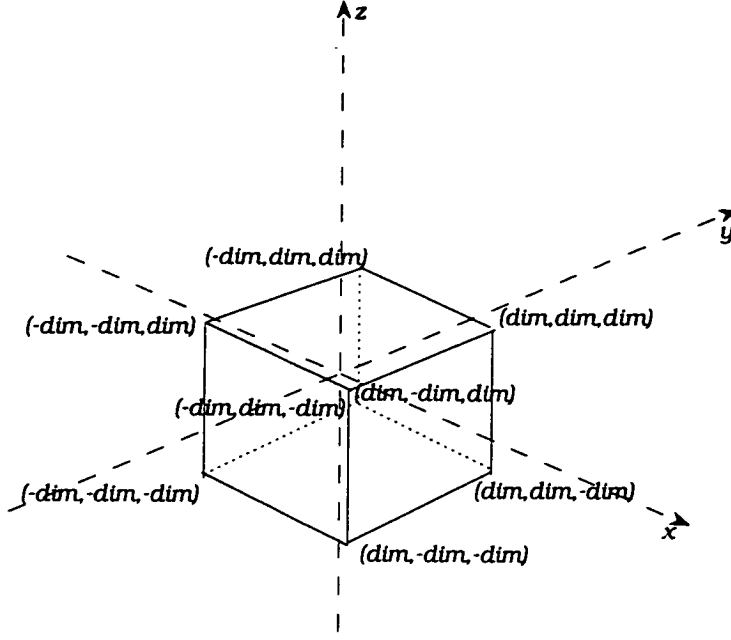


Figure 4.8: Cube of sidelength 2dim centered at the origin

details relating to homogeneous co-ordinates may be found in Appendix A.

For scaling by a factor S_j in direction j

$$\begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

for rotation anti-clockwise by an angle θ about the z-axis

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

for rotation anti-clockwise by an angle θ about the x-axis

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

for rotation anti-clockwise by an angle θ about the y-axis

$$\begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and for translations of T_x, T_y, T_z in each of the x, y , and z directions

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ T_x & T_y & T_z & 1 \end{bmatrix}.$$

The multiplication

$$\begin{bmatrix} T_x & T_y & T_z & 1 \end{bmatrix} = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} a & e & i & 0 \\ b & f & j & 0 \\ c & g & k & 0 \\ d & h & l & 1 \end{bmatrix}$$

reduces to

$$\begin{bmatrix} T_x & T_y & T_z \end{bmatrix} = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} a & e & i \\ b & f & j \\ c & g & k \\ d & h & l \end{bmatrix}.$$

The scaling matrix enables us to magnify or shrink in one or more of the x, y and z directions to give us a rectangular box instead of a cube. Each of the twelve edges of the cube can be represented parametrically by

$$\text{vertex } i + t[\text{vertex } j - \text{vertex } i]$$

where $0 \leq t \leq 1$, the vertices i and j are adjacent and *vertex* i represents the co-ordinates of the vertex i . All that remains to create our perspective drawing, is to find where each viewline to each point on the cube intersects the picture plane. Fortunately, the computer does this for us. As mentioned above, the appropriate program transcript is included in Appendix B, and some resulting perspective drawings of cubes are shown in Figure 4.9.

Now we have constructed our pictures with a given viewpoint, however, if we view this cube not knowing the correct viewpoint, then can we find it? Is this really the perspective drawing of a cube? Could it be a projection of another figure such as a truncated pyramid? Could it be the perspective drawing of a bigger cube placed further away? Discussion of possible ambiguities may be found in Gregory [7, page 35]. It suffices to say that if we do not know the viewpoint or the dimensions of the cube, then the perspective drawings displayed in Figure 4.9 could represent many different cases.

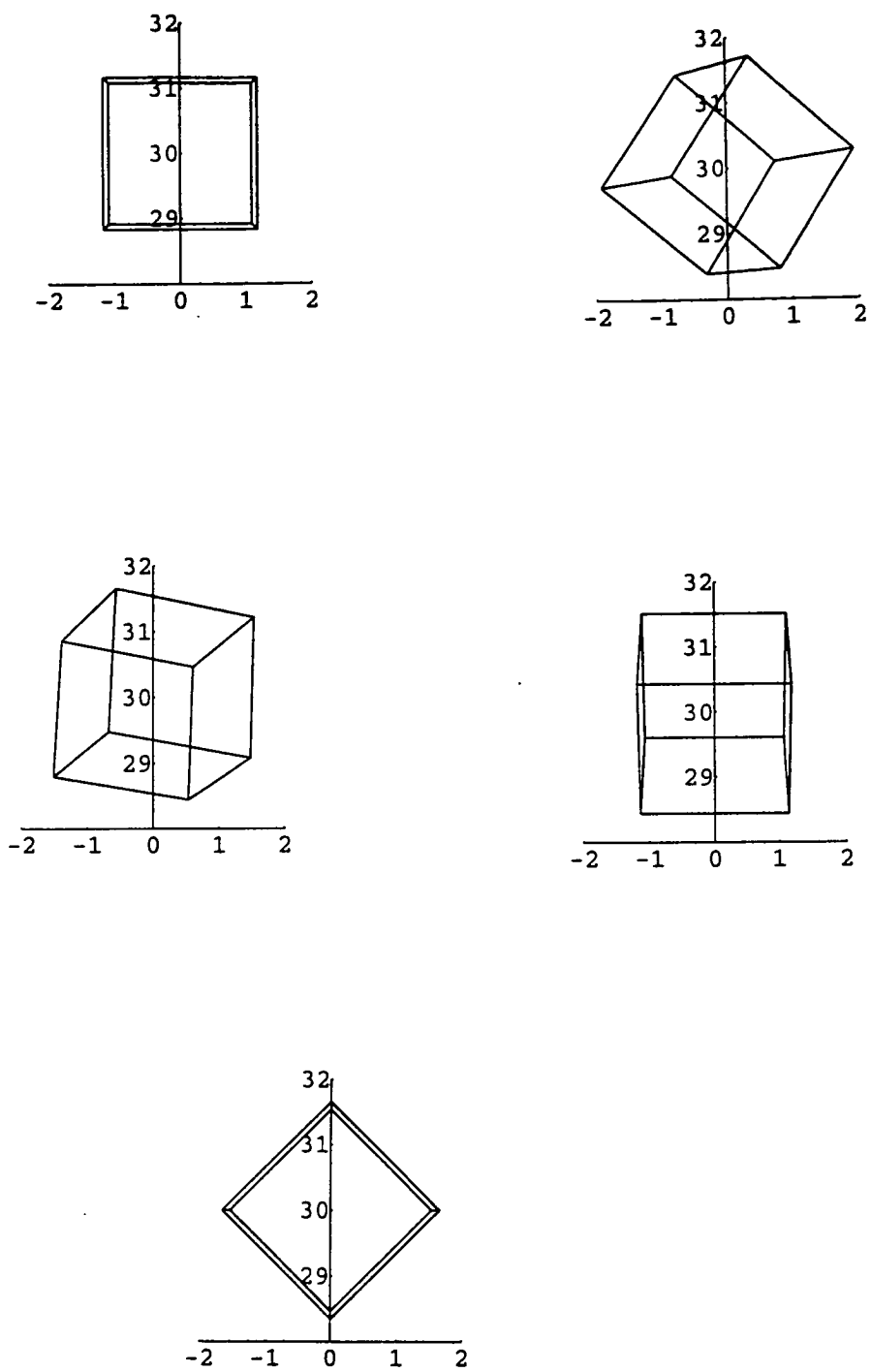


Figure 4.9: Perspective drawings of cubes of similar dimensions, but differing viewpoints.

4.3 Slant Anamorphograms

In Figure 4.5 we saw what could be described as a distorted image of the rectangular grid of squares of Figure 4.4. The apparent distortion occurs when this figure is viewed from the normal viewing position of a page of this thesis, rather than from the correct viewpoint, which is given in the text.

One way to create a slant anamorphic representation of any picture is to cover the picture with a co-ordinatized grid such as the one in Figure 4.4 and then transform the co-ordinates of this picture onto the co-ordinates of the distorted grid. Similar triangle arguments were used by Hickin [11] to describe such transformations.

To create our examples we again use our program of Appendix B. Here any given point, p , of our *correct* picture is transformed to a point, p' , of the distorted picture. By correct picture, we mean one that is easily recognizable provided it is viewed from a given viewpoint. As mentioned earlier, this is not usually the normal viewpoint of a reader of this thesis.

Suppose we want to create a slant anamorphic picture of a cube, then all we need to do is re-create the situation shown diagrammatically in Figure 4.7 (a), where we place the cube about the z -axis. The perspective drawing of such a cube on the plane $y = 0$ is shown in Figure 4.10. The slant anamorphic picture of the same cube is found by creating its perspective drawing on the plane $z = 0$ as shown in Figure 4.11. This latter cube

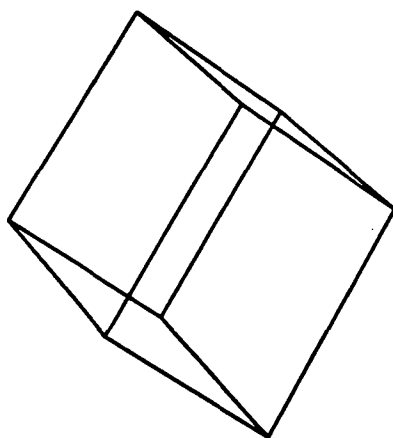


Figure 4.10: Perspective drawing of cube on the plane $y = 0$

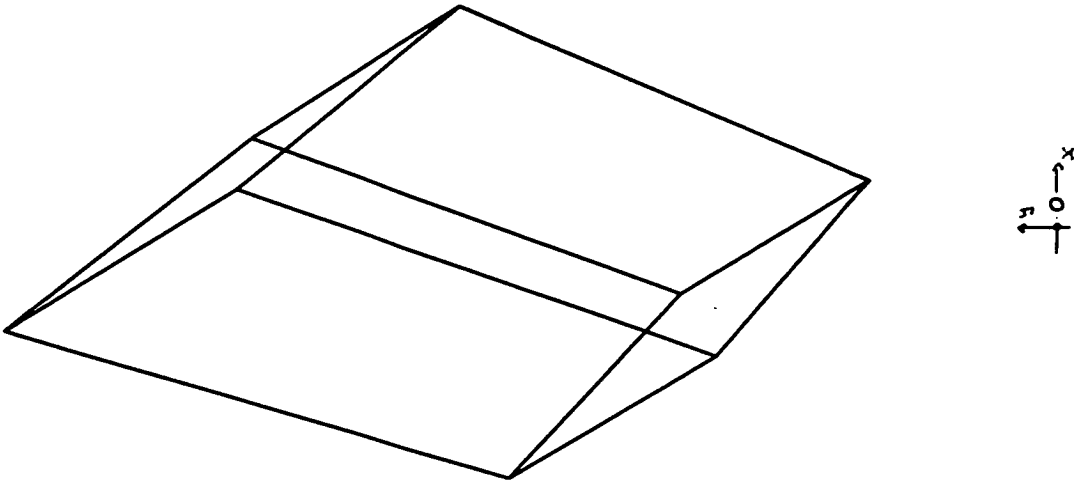


Figure 4.11: Slant anamorphic cube

will not appear distorted if we view it with one eye placed at the position $v = (0, -30, 20)$. This can be approximated by viewing from the side and assuming that the point, o , is at the origin of our co-ordinate system, the page is in the xy -plane, and the z -axis is perpendicular to our page. Figure 4.12 is the perspective drawing on the xz plane of a 7cm cube which has been rotated about the origin as described previously, and then translated by 10cm in the y -direction.

4.4 Curved Surface Anamorphograms

In our perspective drawing examples so far, the picture surface has been planar. In Chapter 2 we alluded to the Fresco art on curved surfaces. We now examine curved surface anamorphograms which will be drawn on a cylinder, cone or sphere. Again the correct viewpoint is vital if we are to perceive a given image.

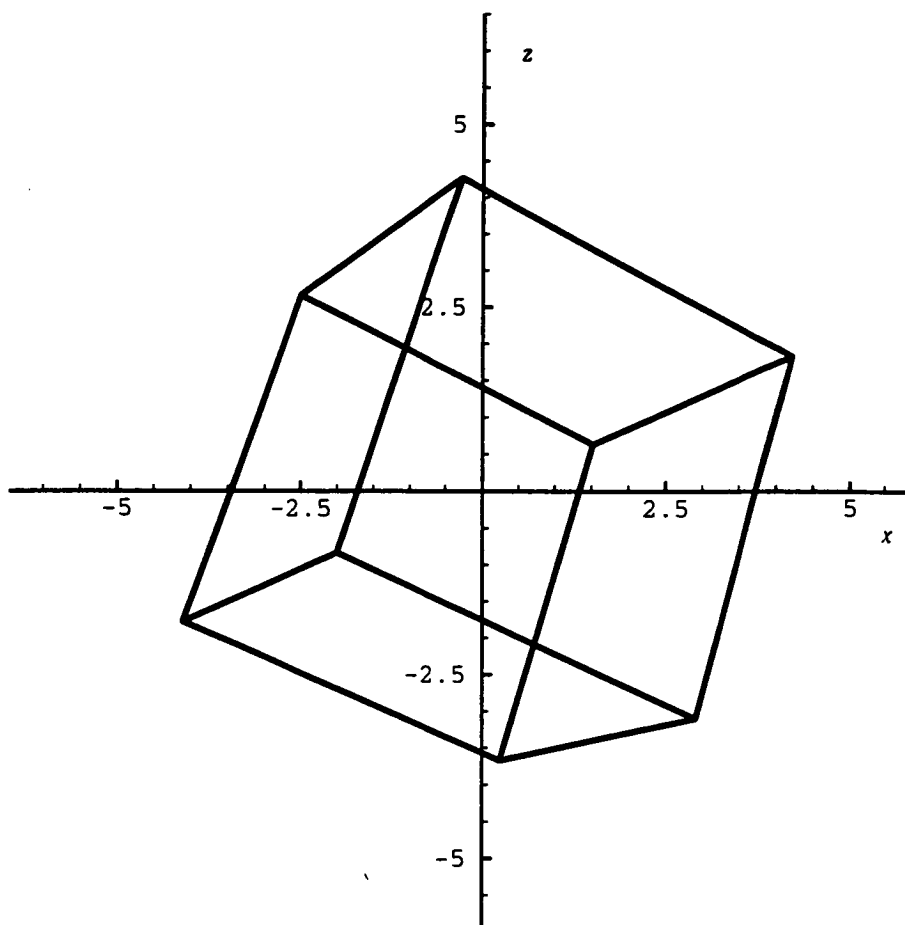


Figure 4.12: Perspective drawing of cube on the xz plane; viewpoint $(0, -30, 0)$

4.4.1 Circular cylinder

Firstly we will consider a circular cylindrical surface. Our method amounts to replacing the picture plane of our perspective drawing by a cylindrical page. That is, we replace the equation representing the picture plane in the program of Appendix B with the equation, $F(x, y, z) = x^2 + y^2 - r^2 = 0$, of the cylinder. Figure 4.13 shows that unless the viewlines are tangential to

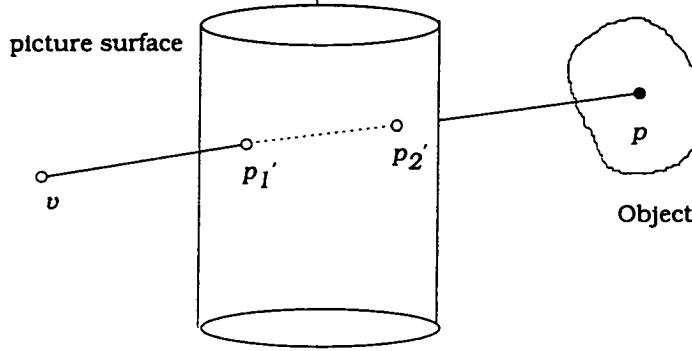


Figure 4.13: Picture surface is a circular cylinder

the cylinder, then they intersect the cylinder at two points. Consequently our program will give us two curved perspective images. These two possibilities, one *near* and the other *far*, are illustrated in Figure 4.14. Depending on the viewpoint, these cross-sections will be parts of ellipses or circles.

In order to obtain individual pictures for each surface, we consider the minimum and maximum values of the parameter, t , which are given to us by finding the intersection points with the cylinder surface of the viewline to any point, $p = (x, y, z)$, on the object. Again the viewline is represented parametrically by

$$\text{viewpoint} + t[p - \text{viewpoint}]$$

and the cylinder has the equation

$$F(x, y, z) = x^2 + y^2 - r^2 = 0.$$

Another consideration is the suitable size of the cylinder in relation to the object being drawn. Figure 4.15 shows that if the object to be drawn is placed behind the cylinder, then it must lie entirely within the region bounded by the cylinder and the tangential viewlines from the eye. The possible size varies according to the position of the viewpoint. The far

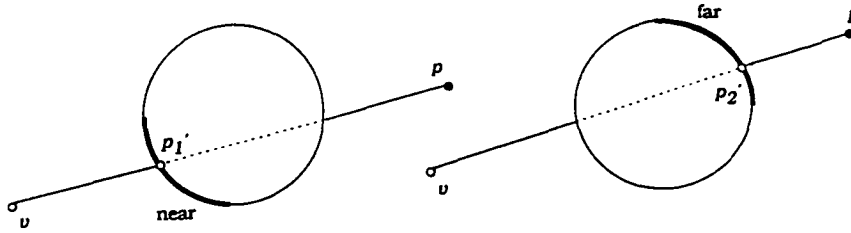


Figure 4.14: Cross-sections viewed from above; near and far

surface allows a larger possible image size if positioned in relation to the front surface as shown. We will apply our program to a cube in various positions to test our results. We can test our results by, firstly wrapping our pictures around a cylinder with correct radius, and then, by viewing from the correct viewpoint. It is probably easier to consider a suitable sized

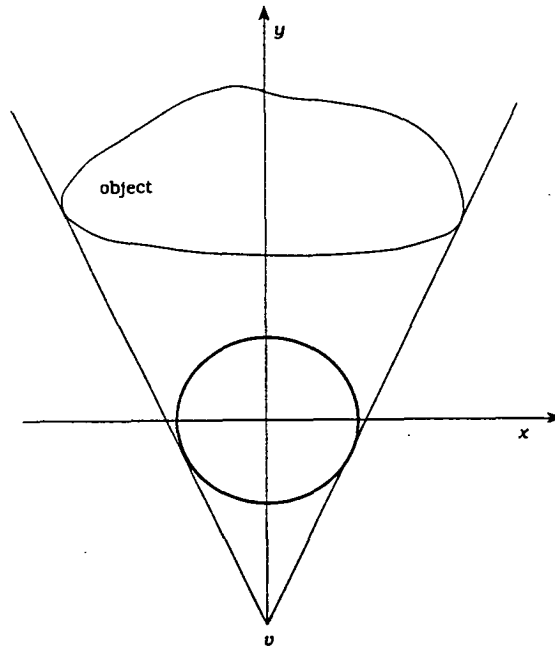


Figure 4.15: Tangential viewlines place bounds on size of object

cylinder given the greatest width and position of the object to be drawn. We can basically use the method for constructing the inscribed circle of a given triangle to find a cylinder of minimum suitable radius.

Figure 4.16 shows the anamorphogram of a cube of sidelength 3 cm. The viewpoint is $(0, -30, 20)$, the cylinder (or picture surface) has equation $x^2 + y^2 - 9 = 0$, and the cylinder lies between the viewpoint and the cube which has been rotated and translated away from the origin. Such relative positions are shown in Figure 4.14. Figure 4.16 (a) shows the case for the surface nearest the viewer, while Figure 4.16 (b), shows the case for the far surface.

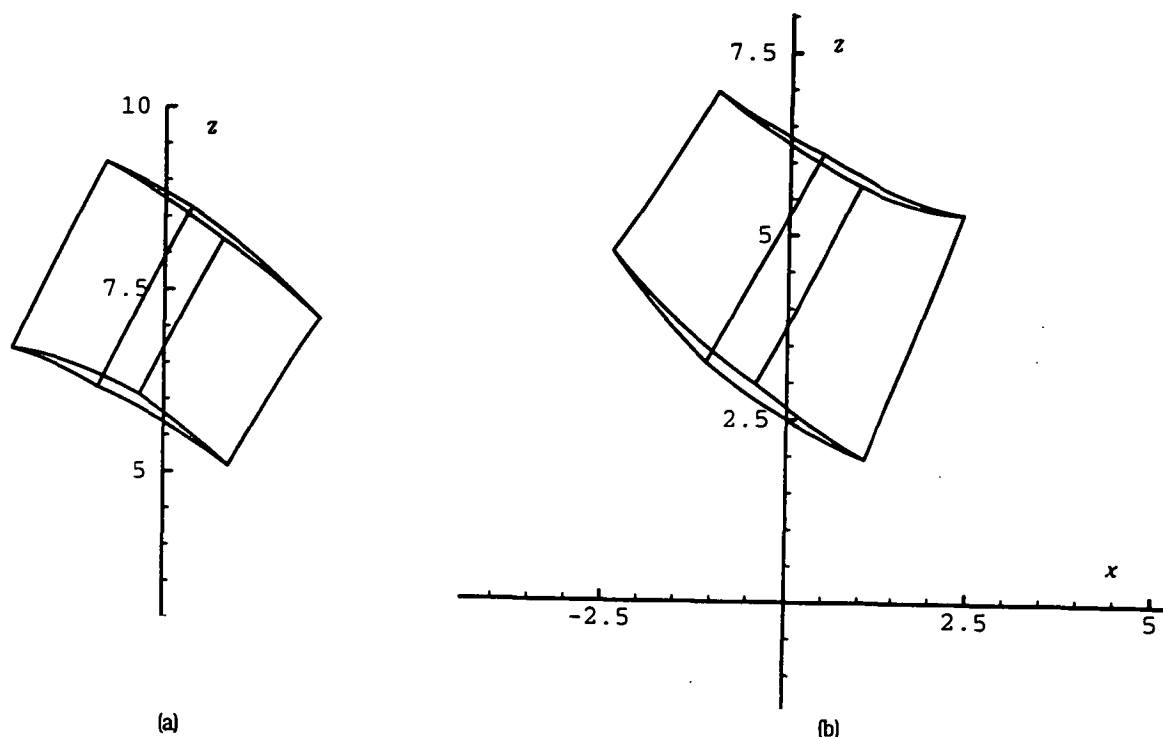


Figure 4.16: (a) Cube for near cylindrical picture surface ; (b) Cube for far cylindrical picture surface

4.4.2 Spherical surface

We have an analogous situation if our picture plane is replaced by a sphere with equation

$$F(x, y, z) = x^2 + y^2 + (z - r)^2 - r^2 = 0.$$

Again we have a projection onto two surfaces, one of which is near, and the other far, as shown in Figure 4.14. Having printed out our pictures on a flat page we need a 'stretchy' material on which to trace these pictures if we want to be able to satisfactorily wrap it around a sphere.

Figure 4.17 shows the anamorphogram of a cube of sidelength 3 cm. The

viewpoint is $(0, 0, 30)$, the sphere (or picture surface) has equation $x^2 + y^2 + (z - 4)^2 - 16 = 0$, and the cube lies between the viewpoint and the sphere. Figure 4.17 (a) shows the case for the near surface, while Figure 4.17 (b), shows the case for the far surface.

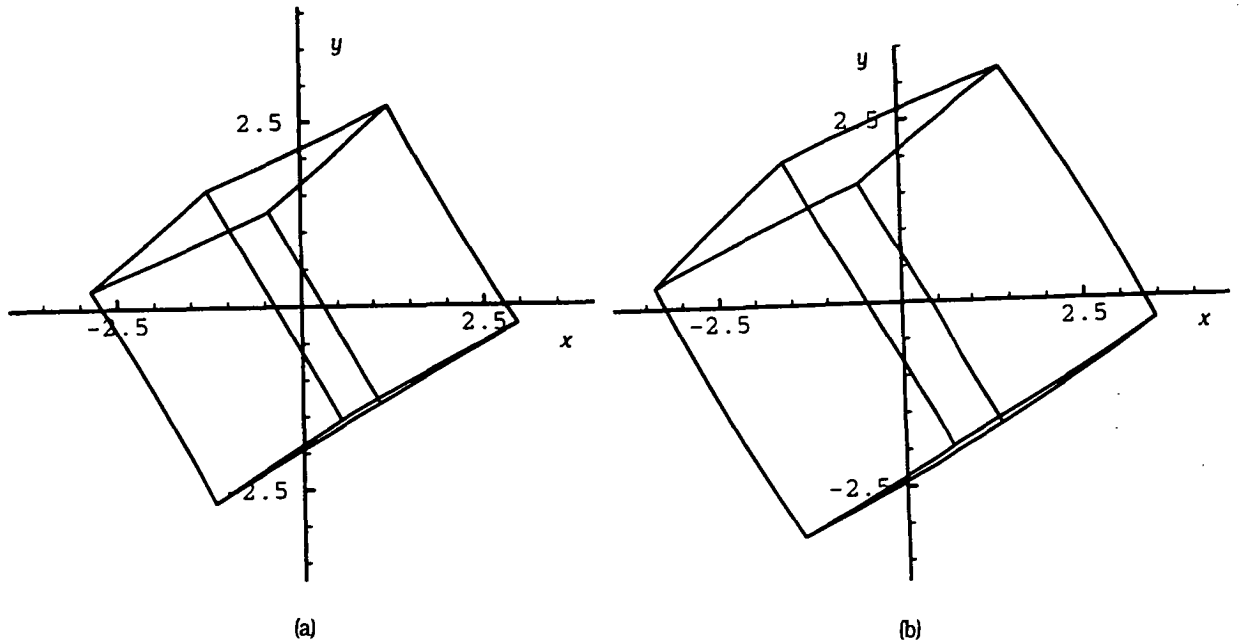


Figure 4.17: (a) Cube for near spherical picture surface ; (b) Cube for far spherical picture surface

4.4.3 Conical surface

For a conical picture surface we again follow a similar method to that for the cylindrical case. Again we simply replace the picture plane by the cone however, we can have a further two situations. The case when the apex of the cone is pointing up, or the case when it is pointing down. The viewpoint could be directly above the apex looking down on the cone or it could be at a point which requires side viewing. Classroom methods for constructing anamorphoses on a cone to be viewed from above are discussed by Hickin [11, page 211]. If we consider Figure 4.18 we can see that if we let the cone be represented by an equation of the form

$$F(x, y, z) = x^2/a^2 + y^2/a^2 - (z - b)^2 = 0$$

which represents a cone with circular horizontal cross-section, then again we

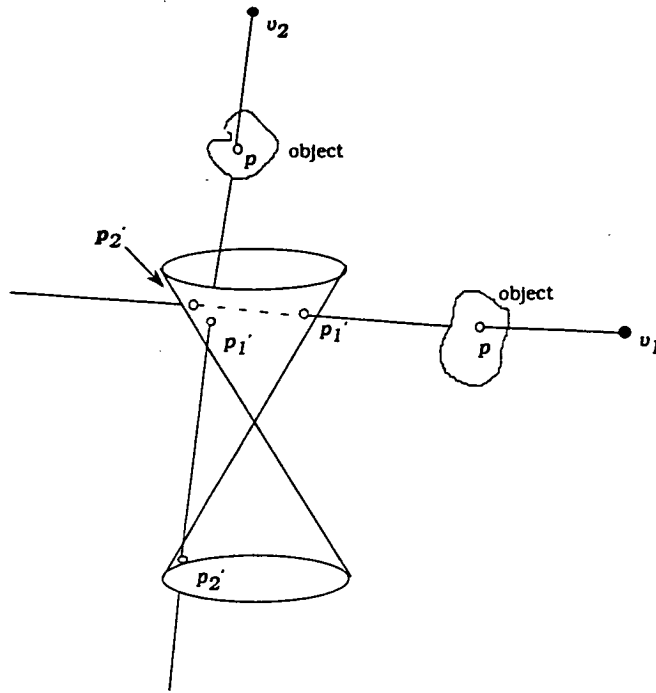


Figure 4.18: Two picture image points for circular cone

have two possible image points p_1' and p_2' on the cone for any object point p . Two possible viewpoints are shown in Figure 4.18. For convenience, we will consider an object placed between the viewer and the cone. Note that the equation of the cone allows us to consider both the cases of cone apex up or down at once, although we will again use our two possible parameter values to consider each part of the cone separately. For a viewpoint on the z-axis we have an analogous situation to v_2 in Figure 4.18 and resulting images of a 3cm cube such as those in Figure 4.19. For a viewpoint such as v_1 our images are such as those in Figure 4.20.

4.5 Mirror Anamorphograms

We will now consider the problem of creating a distorted picture which we want to 'unscramble' by viewing its reflection in a mirror. This mirror will be planar, cylindrical, conical or spherical. We begin by looking at mirrors in general.

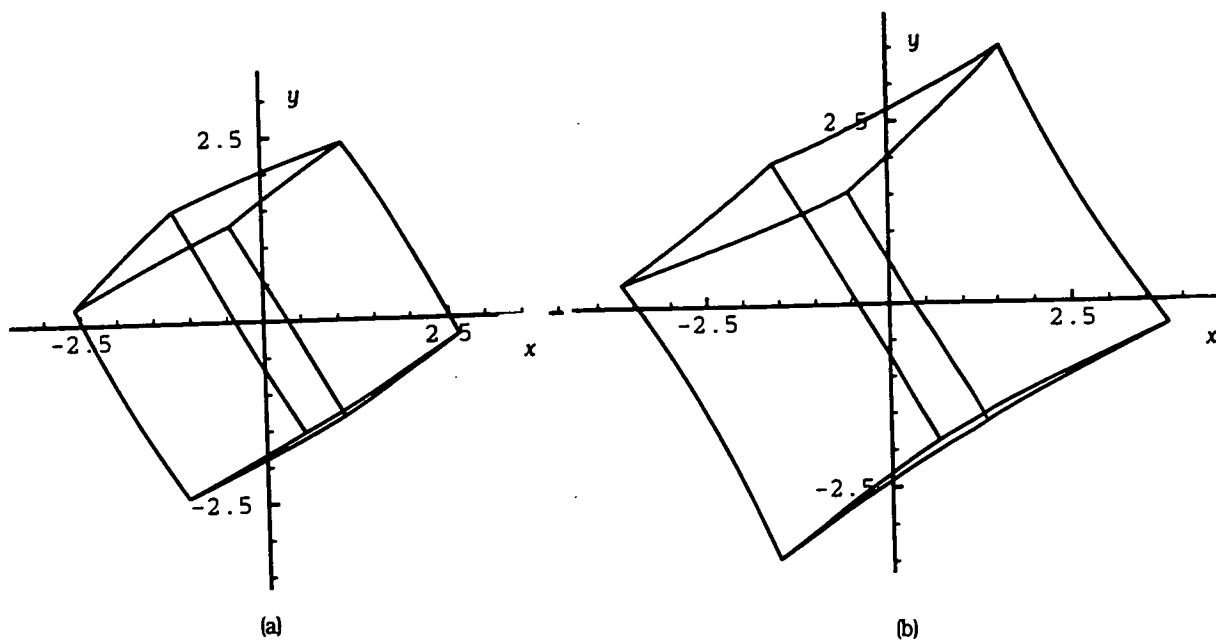


Figure 4.19: Viewpoint $(0, 0, 30)$, cone $x^2 + y^2 - 1/3z^2 = 0$, (a) inside top of conical picture surface (near), (b) outside far section of cone

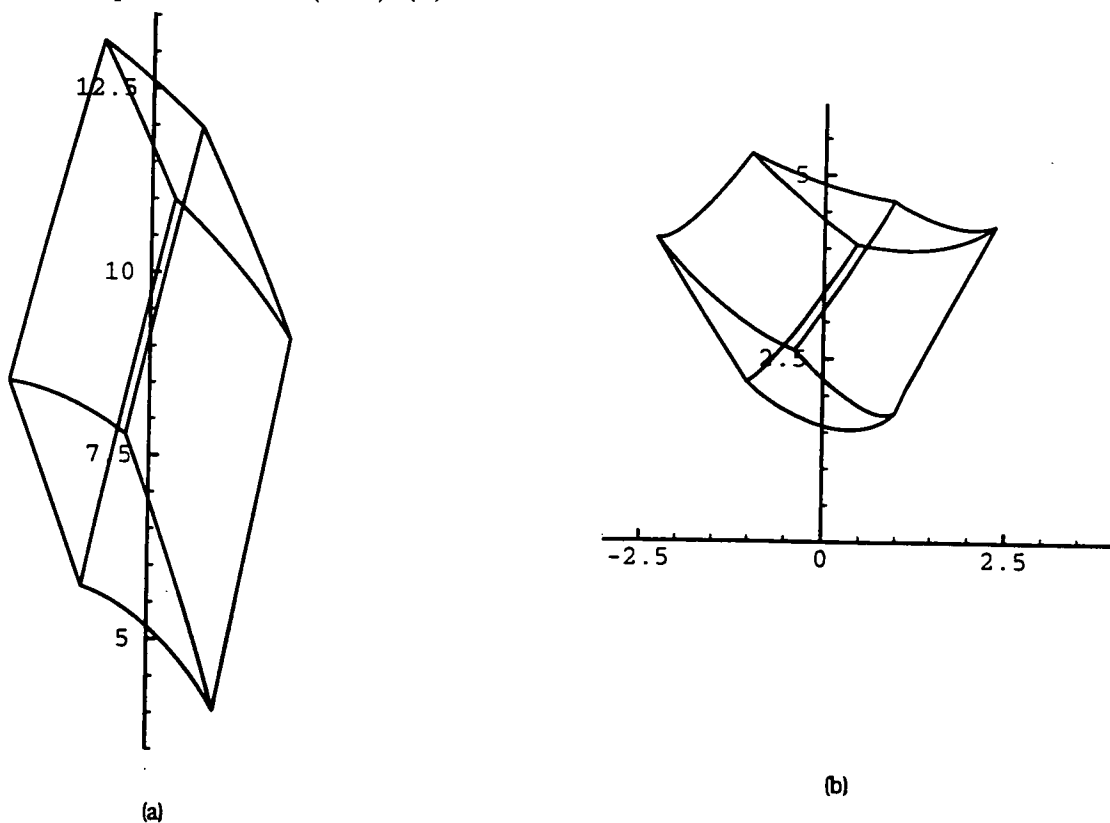


Figure 4.20: Viewpoint $(0, -30, 30)$, cone $x^2 + y^2 - z^2 = 0$, (a) outside top of conical picture surface (near), (b) inside far section of the top section of cone

4.5.1 General mirror

Firstly, a very brief description of one of the fundamental laws of optics is required. When a ray of light strikes any boundary between two transparent substances in which the velocity of light is different, it is in general divided into a reflected and a refracted ray. Consider Figure 4.21 and let $p \vee m$ represent the incident ray, and let it make an angle ϕ with N_m , the normal to the surface at m . This angle, ϕ , is called the *angle of incidence* and the plane defined by $(p \vee m)$, and the normal, N_m , is called the *plane of incidence*.

The Law of Reflection can be stated as:

The reflected ray lies in the plane of incidence, and the angle of reflection equals the angle of incidence.

That is, $p \vee m$, the reflected ray $m \vee v$, and the normal to the surface at m , are all in the same plane and $\phi = \phi'$.

Now suppose we have the situation shown in Figure 4.22 where $v =$

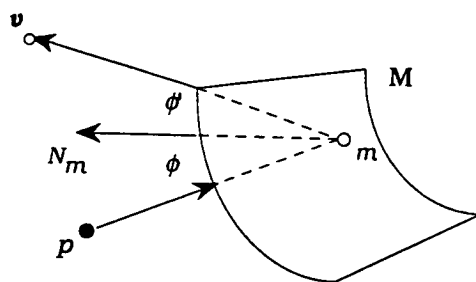


Figure 4.21: Angle of incidence = Angle of reflection

(v_x, v_y, v_z) represents the viewpoint, M represents the surface of the mirror, $p' = (x', y', z')$ is an image point (viewed in the mirror) of an object point $p = (x, y, z)$, \vec{N}_m represents the direction vector of the normal to the surface at the point $m = (m_x, m_y, m_z)$, $v \vee m$ represents the incident ray. The angle of incidence, ϕ , equals the angle of reflection.

If we are given v , p' , the equation of M and the equation of the plane of p , then we proceed to find the point, p , where the reflected ray meets this plane. The viewline from the viewpoint to the perceived image, $p' = (x', y', z')$, in

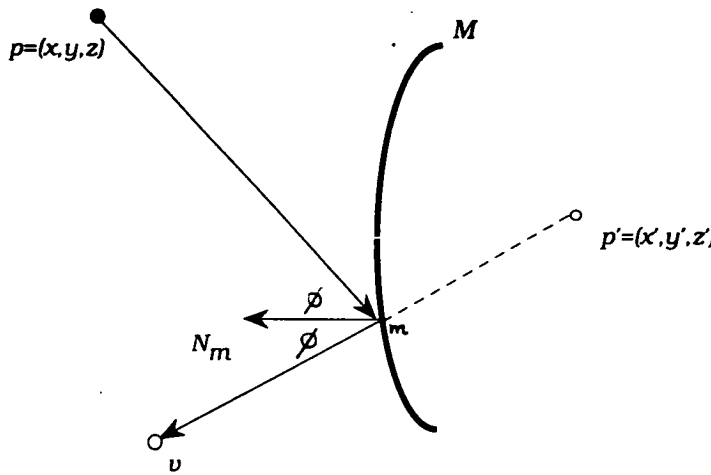


Figure 4.22: Viewer, image-object connections

the mirror is given by

$$(v_x, v_y, v_z) + t * [(x', y', z') - (v_x, v_y, v_z)]. \quad (4.1)$$

Next we find the intersection point, m , of this line with the mirror's surface which has equation $F(x, y, z) = 0$, say. To find the normal to the surface at m we calculate the gradient of F at this point, $\nabla F[(m_x, m_y, m_z)]$, and this is perpendicular to the tangent plane.

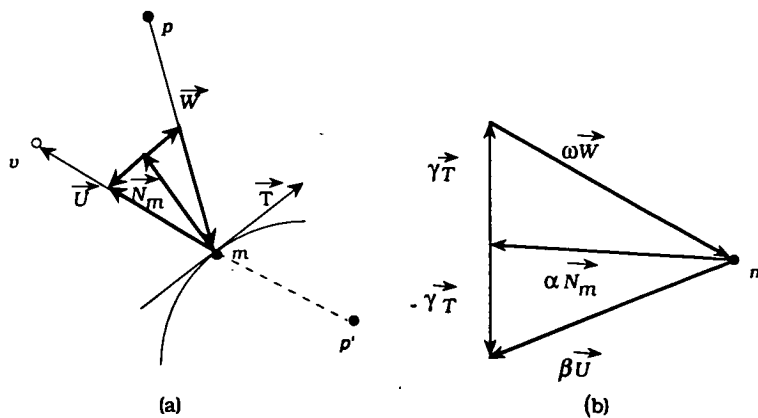


Figure 4.23: Vector representation of triangles on plane of incidence.

Now if we consider the co-planar vectors at the surface intersection point, m , we have the situation shown in Figure 4.23. Consider the two congruent triangles, of Figure 4.23 (b), where \vec{T} is the tangent vector at m , \vec{N}_m is the direction vector of the normal vector, \vec{U} is the direction vector of the viewline to any image point and \vec{W} is the direction vector of the reflected line from m to the object. Now for the two triangles vector addition gives

$$-\omega \vec{W} = \alpha \vec{N}_m + \gamma \vec{T}$$

$$\beta \vec{U} = \alpha \vec{N}_m - \gamma \vec{T},$$

where $\omega, \alpha, \beta, \gamma$ are scalars and combining gives

$$-\omega \vec{W} + \beta \vec{U} = 2\alpha \vec{N}_m.$$

Now $\alpha \vec{N}_m$ is the projection of $\beta \vec{U}$ on \vec{N}_m and so

$$\alpha \vec{N}_m = \frac{\vec{N}_m \cdot \beta \vec{U}}{\vec{N}_m \cdot \vec{N}_m} \vec{N}_m.$$

Hence by combining the latter two results we have

$$\Omega \vec{W} = \vec{U} - 2 \frac{\vec{N}_m \cdot \vec{U}}{\vec{N}_m \cdot \vec{N}_m} \vec{N}_m, \quad (4.2)$$

where Ω is a scalar. Now the parametric representation of the reflected line $m \vee o$ is

$$m + t\vec{W},$$

which we can express in terms of the viewline vector and the normal vector by application of equation 4.2.

In creating a distorted picture of our rectangular grid or cube, it is this anamorphogram which will be our ‘object’ of the above discussion. Our object surface will be our picture page which will lie on the xy-plane of our co-ordinate system. Our image point, p' , in the mirror will be a point of the rectangular grid, or a point of an edge of a cube. The task is to find an object point, $p = (x, y, z)$, for any image point, $p' = (x', y', z')$, in the mirror. All we need to find p , is the intersection point of a reflected line, such as $m \vee p$ of Figure 4.22, with our picture page plane. In fact we hope that this intersection point is actually on our picture page which means that a suitable selection of viewpoint is essential. In contrast to our preceding examples, p represents a point on the distorted picture and p' represents a point of the ‘correct’ image viewed in the mirror. The program which allows us to create such anamorphograms using the above results of subsection 4.5.1 may be found in Appendix D.

4.5.2 Cylindrical mirror

Suppose our cylindrical mirror has equation

$$F(x, y, z) = x^2 + y^2 - r^2 = 0. \quad (4.3)$$

Figure 1 consists of two diagrams. Diagram (a) shows a cylindrical mirror with a grid pattern on its surface. A point p is marked on the grid. A dashed line represents the optical axis, and a coordinate system with x , y , and z axes is shown. Diagram (b) shows a cylindrical mirror with a point $p' = (x', y', z')$ on its surface. A point $p_2 = (x_2, y_2, z_2)$ is marked on the surface. A dashed line represents the optical axis, and a coordinate system with x , y , and z axes is shown. The label "cylindrical mirror" is present in both diagrams.

and maximum values of the parameter t given by solving the equations 4.3 and 4.1. This parameter enables us to find the intersection point on each mirror surface where the principal ray is reflected to our eye from the object. This corresponds to the point m in our discussion in the previous subsection 4.5.1. Some resulting pictures of our grid and cube to be viewed in our near surface mirror are shown in Figure 4.25. In these examples, our viewpoint was $(0, -30, 30)$, the cylindrical mirror had cross-sectional diameter 6cm, and the grid could be seen as a reflection on the xz -plane. The cube with sidelength 1.5cm was placed about the z -axis. The resulting anamorphograms (distorted pictures) lie on the xy -plane which contains our page. It is to be noted that co-ordinatization of corresponding grids allows the creation of anamorphic drawings by hand as described in Leeman, Elffers and Schuyt [15], Gardner [6], Hamngren [9] and Hickin [11].

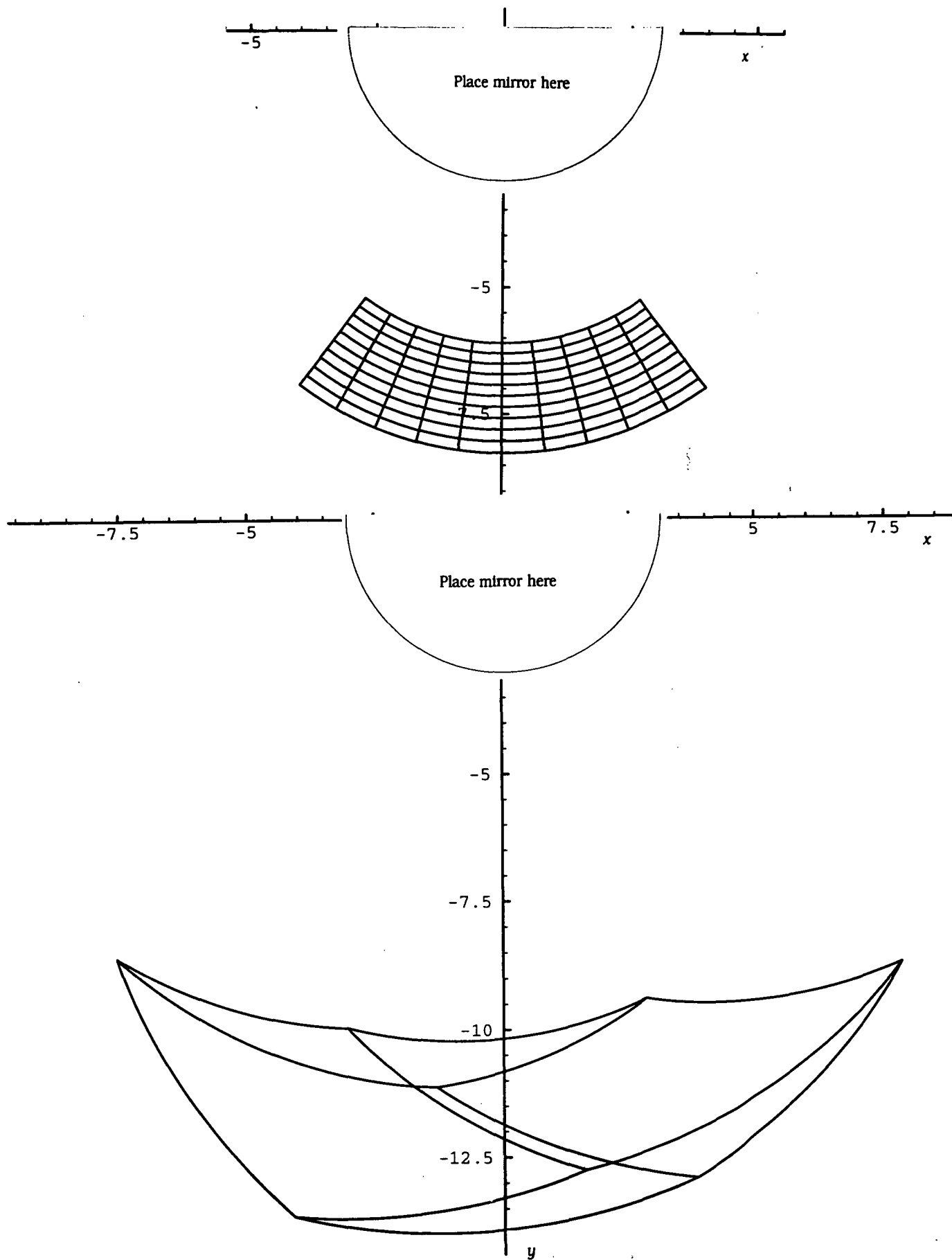


Figure 4.25: Anamorphograms of grid and cube for viewing in near cylindrical mirror $x^2 + y^2 - 9 = 0$; viewpoint $(0, -30, 30)$

4.5.3 Conical mirror

Here we have an analogous situation to that discussed in subsection 4.4.3 except that our cone surface is mirrored. The equation of this mirror surface has the form

$$F(x, y, z) = x^2/a^2 + y^2/a^2 - (z - b)^2 = 0.$$

Our viewpoint must be chosen carefully so that our reflected rays to the object in the xy - plane will intersect our picture page. Various viewpoint scenarios are represented in Figure 4.26. Figure 4.27 shows an anamor-

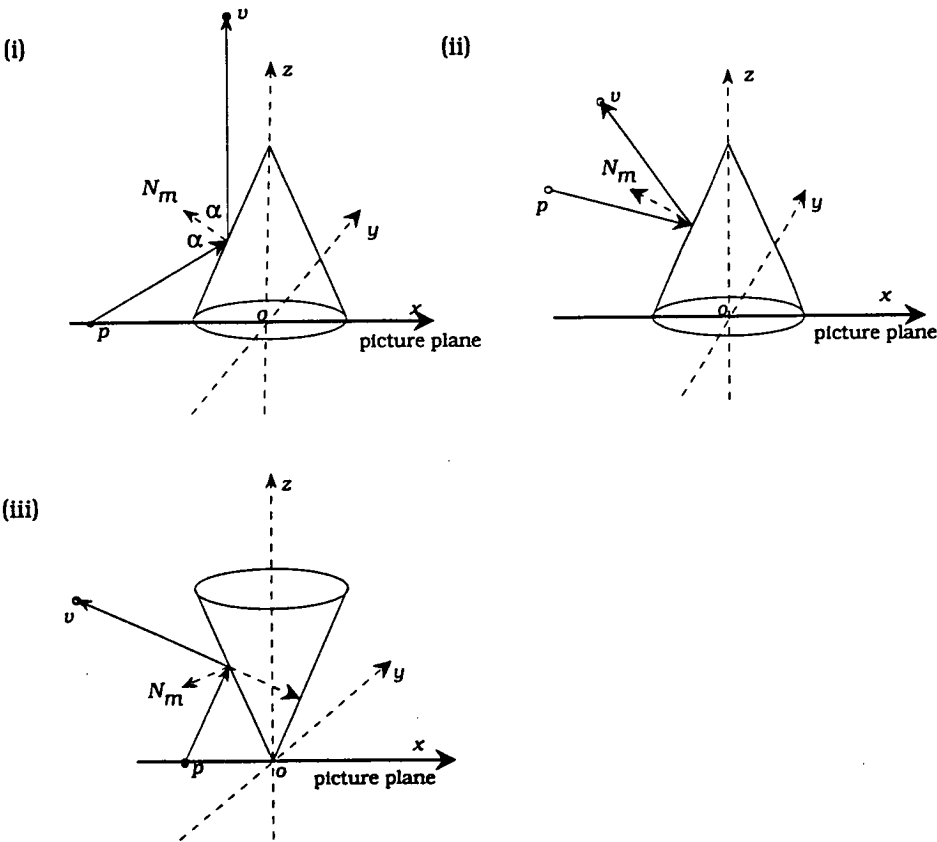


Figure 4.26: Viewpoint v , object p , normal N_m

phogram for a rectangular grid where the viewpoint is $(0, 0, 15)$ and the cone $(x^2 + y^2 = 1/3(z - 10)^2)$ sits on the xy plane at the origin, as shown in

(i) and (ii) of Figure 4.26. Figure 4.28 shows the anamorphogram of a cube of sidelength 2cm. Again the viewpoint is $(0, 0, 15)$.

Care needs to be taken if part of our image is parallel to the axes and in particular includes the origin. This is most likely linked with inversion with respect to circles. Hinkin [11] gives a transformation formula in terms of polar co-ordinates. Kuchel [14] describes how to produce a cone which gives a true circle inversion.

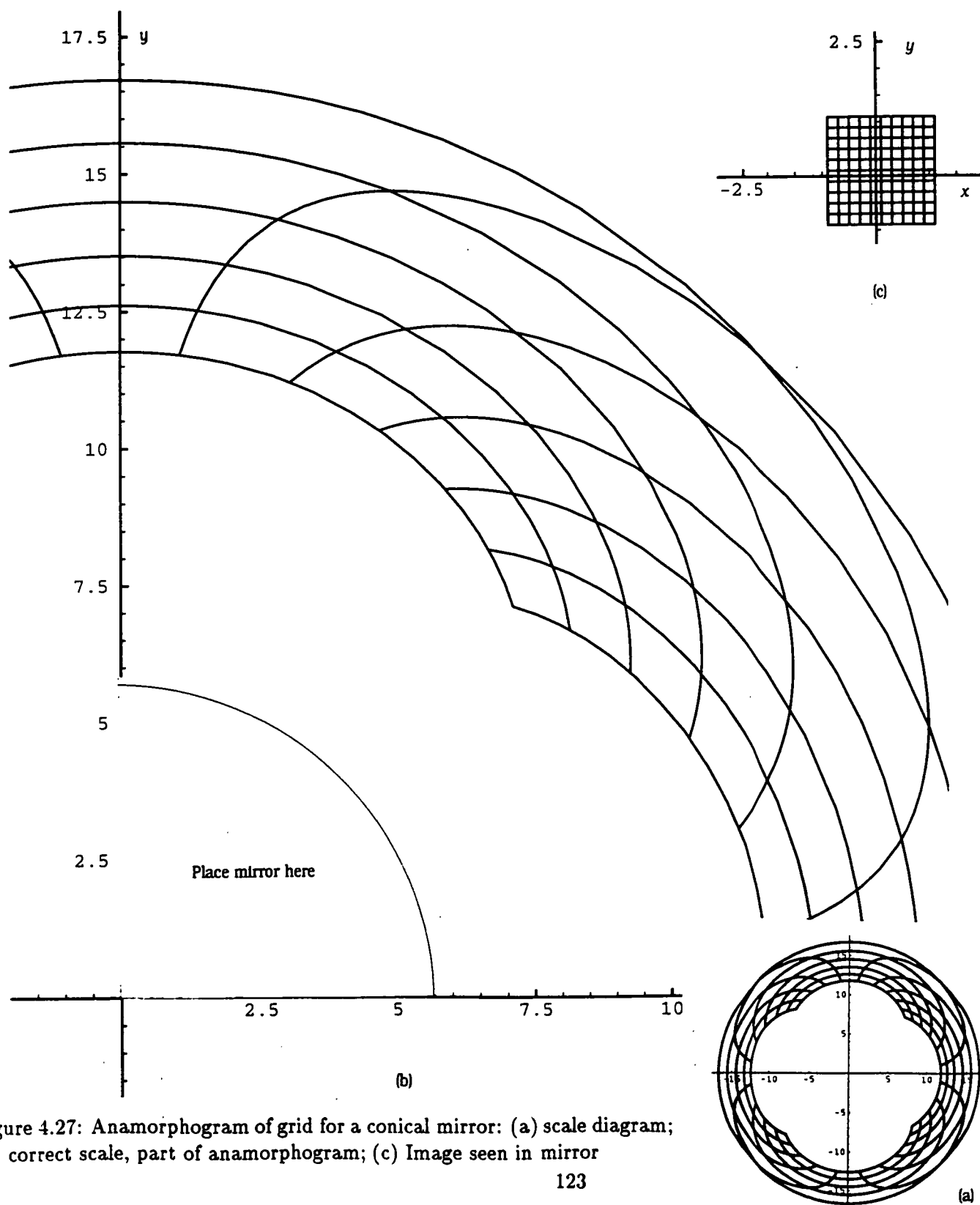


Figure 4.27: Anamorphogram of grid for a conical mirror: (a) scale diagram; (b) correct scale, part of anamorphogram; (c) Image seen in mirror

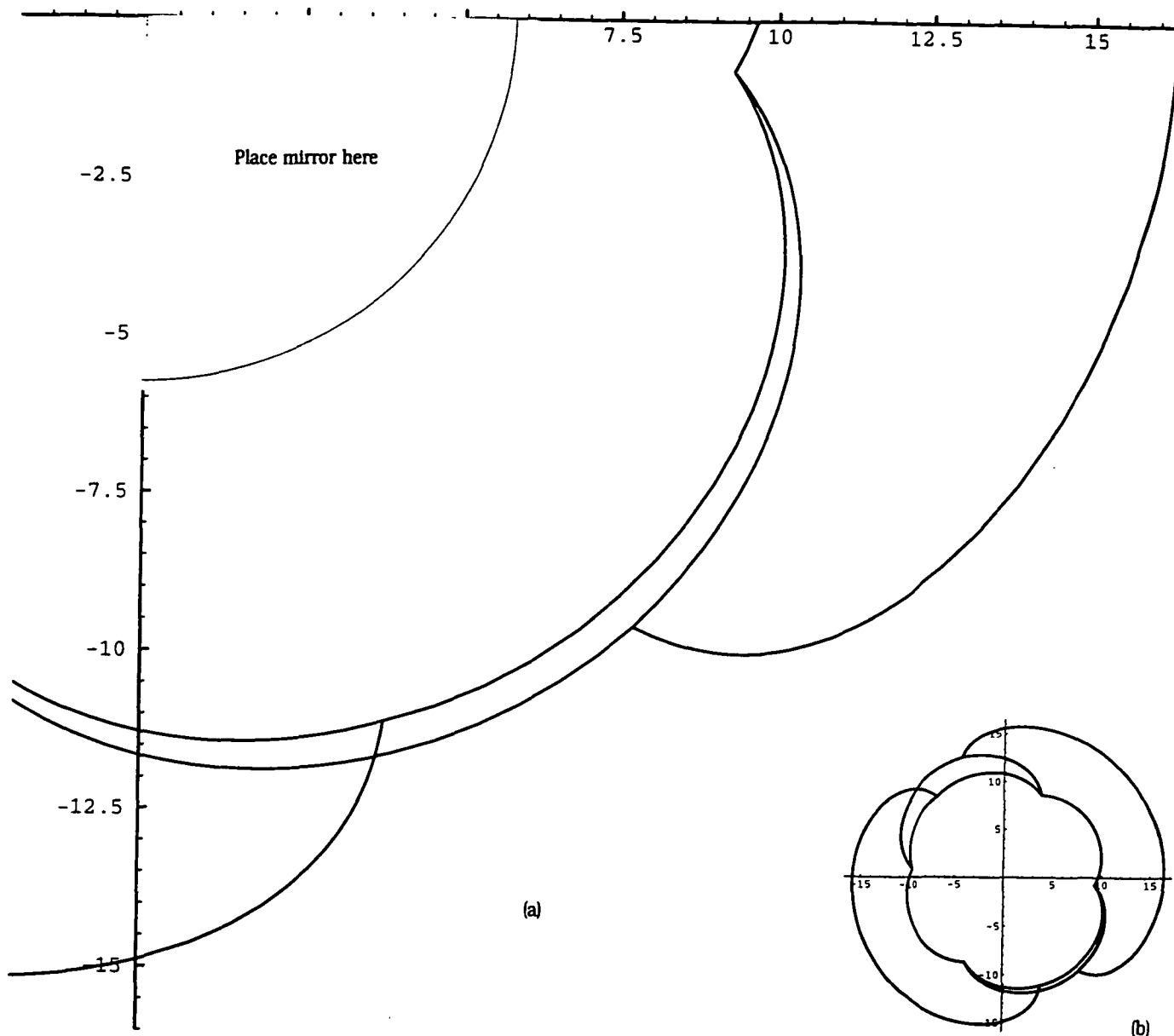


Figure 4.28: Anamorphogram of cube for conical mirror: (a) correct scale, part of anamorphogram; (b) scale diagram

4.5.4 Spherical mirror

We consider a sphere with equation of the form

$$F(x, y, z) = x^2 + y^2 + (z - r)^2 - r^2 = 0,$$

which will sit on our picture page at the origin of our co-ordinate axes. Again care must be taken with the positioning of the viewpoint relative to the sphere. This will be affected by the sphere dimensions.

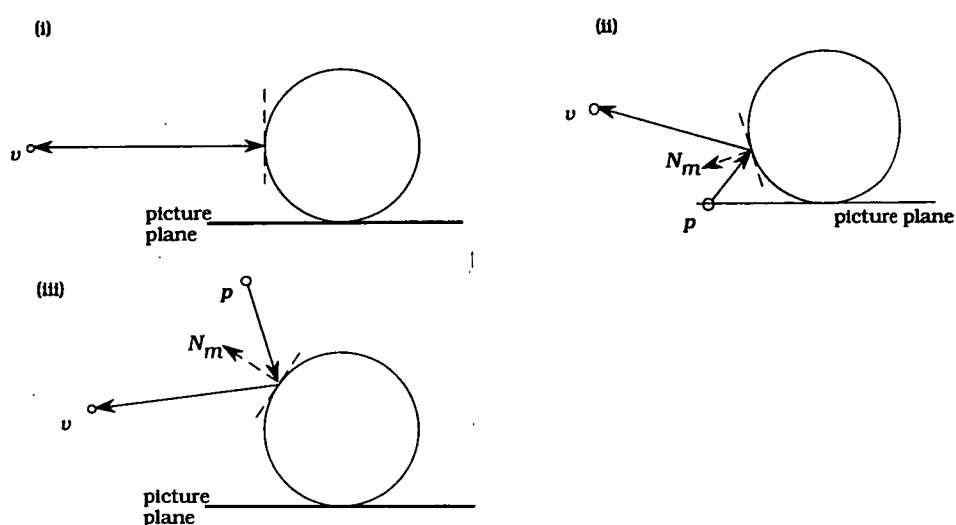


Figure 4.29: Viewpoint v , object p , normal N_m

Figure 4.29 represents various viewpoints and it can be seen that in cases (i) and (iii) we would not see the reflection of any object on our picture page. The anamorphogram, on the xy plane, of a rectangular grid is shown in Figure 4.30. Here the mirror is spherical with radius 6 cm, and the viewpoint is $(0, -30, 5)$. The anamorphogram of a cube of sidelength 2cm is shown in Figure 4.31. The cube has been rotated about the origin and then translated 2.5cm in the z direction. *If we view such an anamorphogram with a mirror of different curvature from that for which it was designed, we expect some distortion.*

4.6 Binocular Examples

All our examples so far have been constructed by considering only one viewpoint, however in reality most of us see the world around us with two eyes. In Chapter 3 we discussed how the horizontal separation of our eyes means that they individually view our surroundings from different viewpoints. The

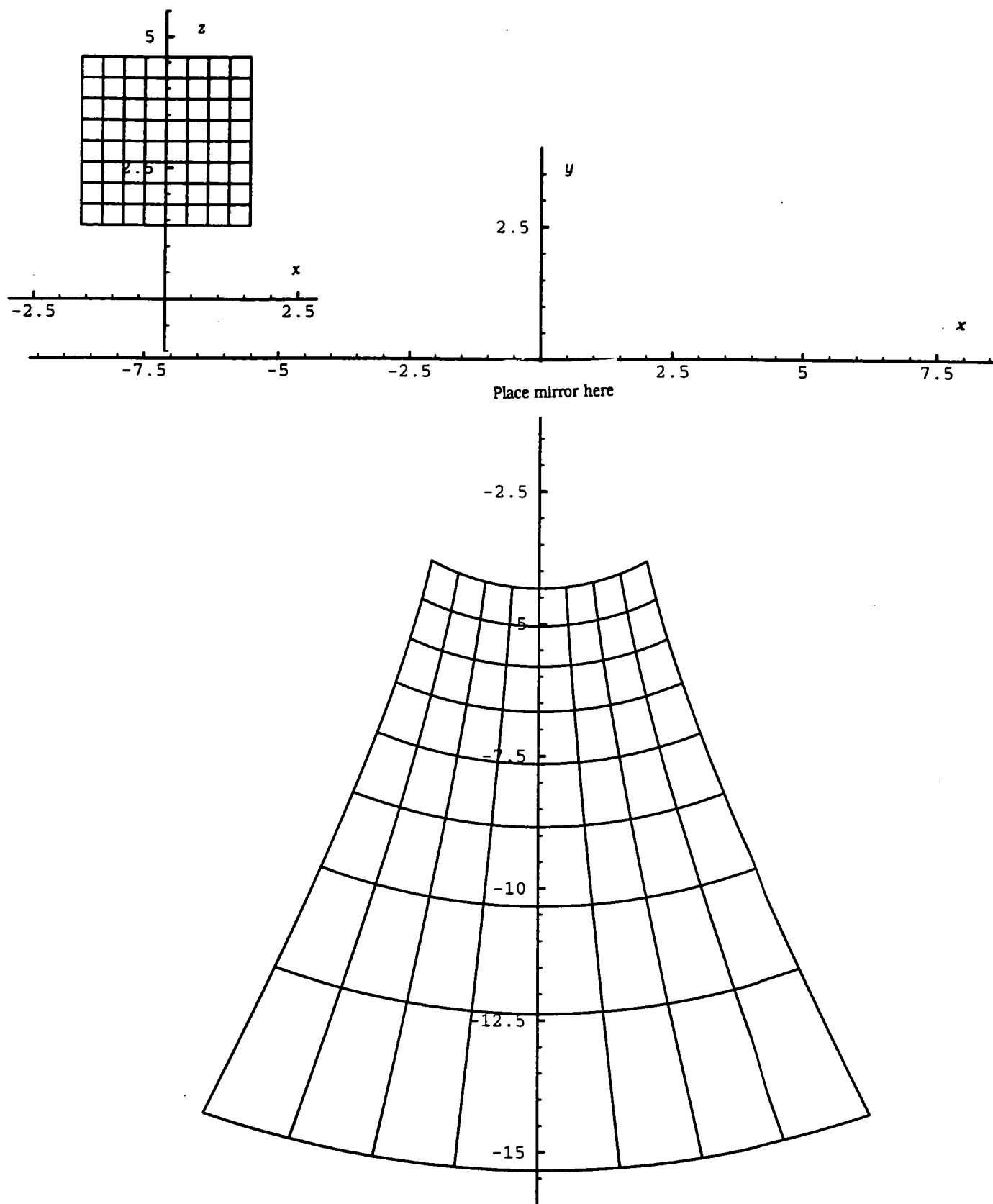


Figure 4.30: Spherical mirror anamorphogram of the rectangular grid shown in the inset.

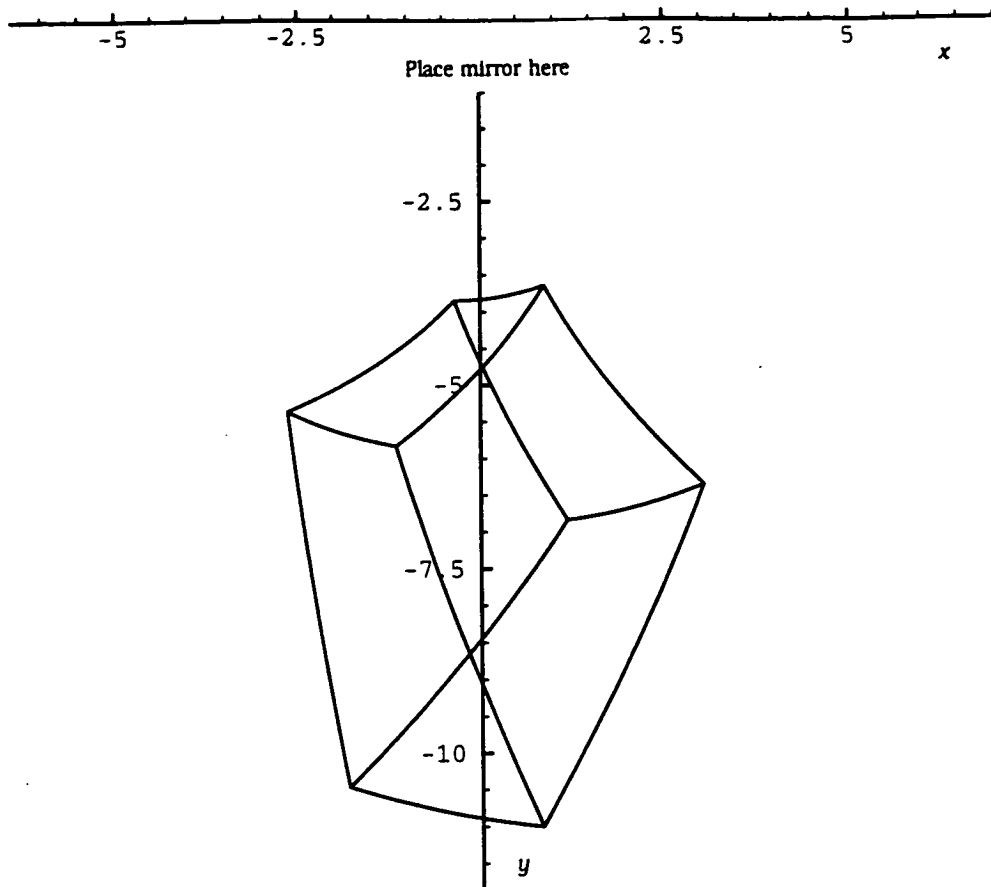


Figure 4.31: Spherical mirror anamorphogram of a cube

two separate images are cleverly combined by our brains to enable us to see differences in depth, and hence, a three-dimensional world. The two different viewpoints which imply retinal disparity were described as the most vital feature for depth perception.

4.6.1 Anaglyphs

We will now *re-view* some of our preceding examples representing a cube, but this time, when viewed correctly, our two-dimensional drawings will present a very vivid three-dimensional image. This will be achieved by creating anaglyphs for each example. The idea of such a technique was introduced in Chapter 2.

Anaglyphs are pictures of an object (or scene) which present two views of the object; one for each eye. In our cases, each of the views will be a perspective drawing of a cube. Each picture is a different colour (traditionally red and green or blue) to enable selective viewing. This is achieved by the viewer wearing glasses with appropriately coloured 'lenses' or filters so that each eye 'sees' only its own picture. (Green lens promotes view of the red picture and vice-versa). The two images are then cleverly combined by the

brain so that we perceive a 3-dimensional picture. The eye separation of this author is 5.75cm (measured by an optometrist using a pupilometer) and so the following pictures were created for viewpoints of both $(-2.875, -30, 0)$ and $(2.875, -30, 0)$. These are distinct from the single viewpoint $(0, -30, 0)$ used in the case for monocular viewing. In every case, similar changes were made to the original viewpoint. It must be noted again that all the viewpoints chosen for the monocular cases ensured that the anamorphogram was on our page.

In the following examples we have used the same cube dimensions and positions as in the corresponding monocular examples considered earlier. Figure 4.32 is the anaglyph of the cube in Figure 4.12. Figure 4.33 (a) and (b) are

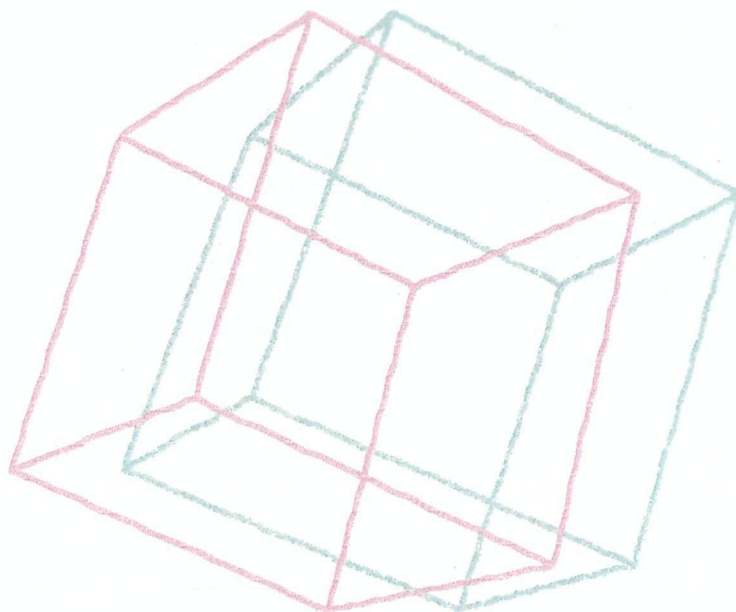


Figure 4.32: Perspective drawings of cube: green image for viewpoint $(2.875, -30, 0)$, red image for viewpoint $(-2.875, -30, 0)$.

the anaglyphs of the cubes of Figure 4.16 (a) and (b) respectively.

We also include one mirror example to be viewed in the outside of a cylindrical mirror. Figure 4.34 is the anaglyph for the cube of Figure 4.25. An interesting experiment in viewing anaglyphs, is to reverse the images for each eye by turning the viewing 'spectacles' around. In this case, the right eye views the left eye's image and vice-versa. In Chapter 3 we discussed the 'cross-eyed' technique for viewing dot stereograms which amounts to

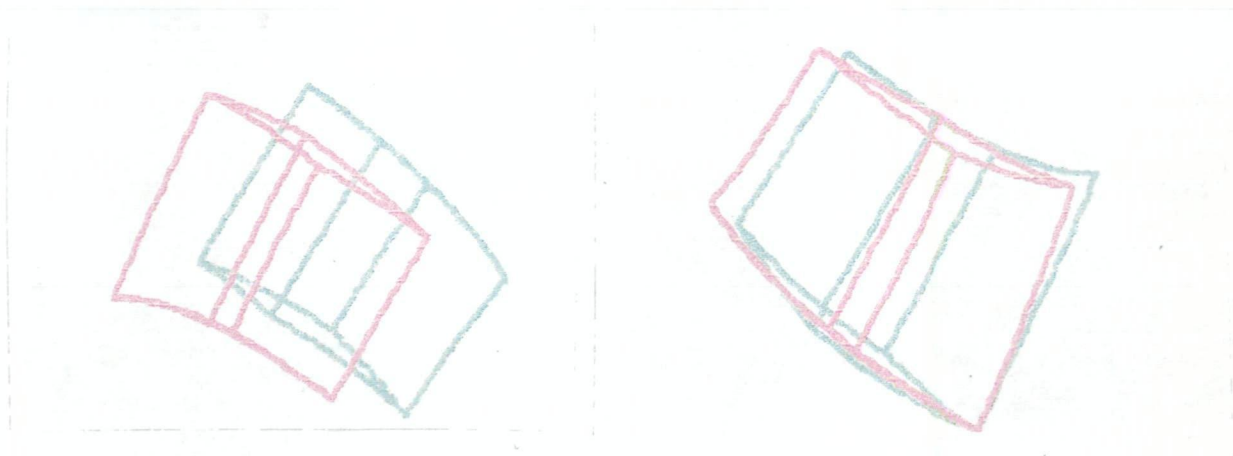


Figure 4.33: Anaglyph for cylindrical surface anamorphograms of a cube

exactly this process of exchanging the images. We found that the ‘perceived image’ in this case appears to lie between the viewer and the stereogram page rather than behind the page. We obtain similar results in viewing the anaglyphs included here. In Chapter 6 we consider in some detail, the mathematical properties of the reversed perceived images for more complicated stereograms, than those of Chapter 3.

4.6.2 Revisiting Luneburg’s experiment

Having considered binocular vision, slant anamorphograms and curved surface anamorphograms, we are now in a position to further examine Luneburg’s experiment of subsection 3.5.2.

To re-iterate:

In this experiment we constructed sub-pencils of lines through two points, l and r , which represented the nodal points of the eyes. The angle between neighbouring lines remains constant. We then viewed these pencils of lines from viewpoints directly above l and r , as illustrated in Figure 4.35. We note that this situation is reminiscent of the one represented in Figure 4.5 when we viewed a grid on the plane $z = 0$, from a point above, and near its side. In that case we viewed with one eye only and we referred to the distorted grid as a slant anamorphogram. In this case, our sub-pencils of lines could both be considered as examples of slant anamorphograms; one

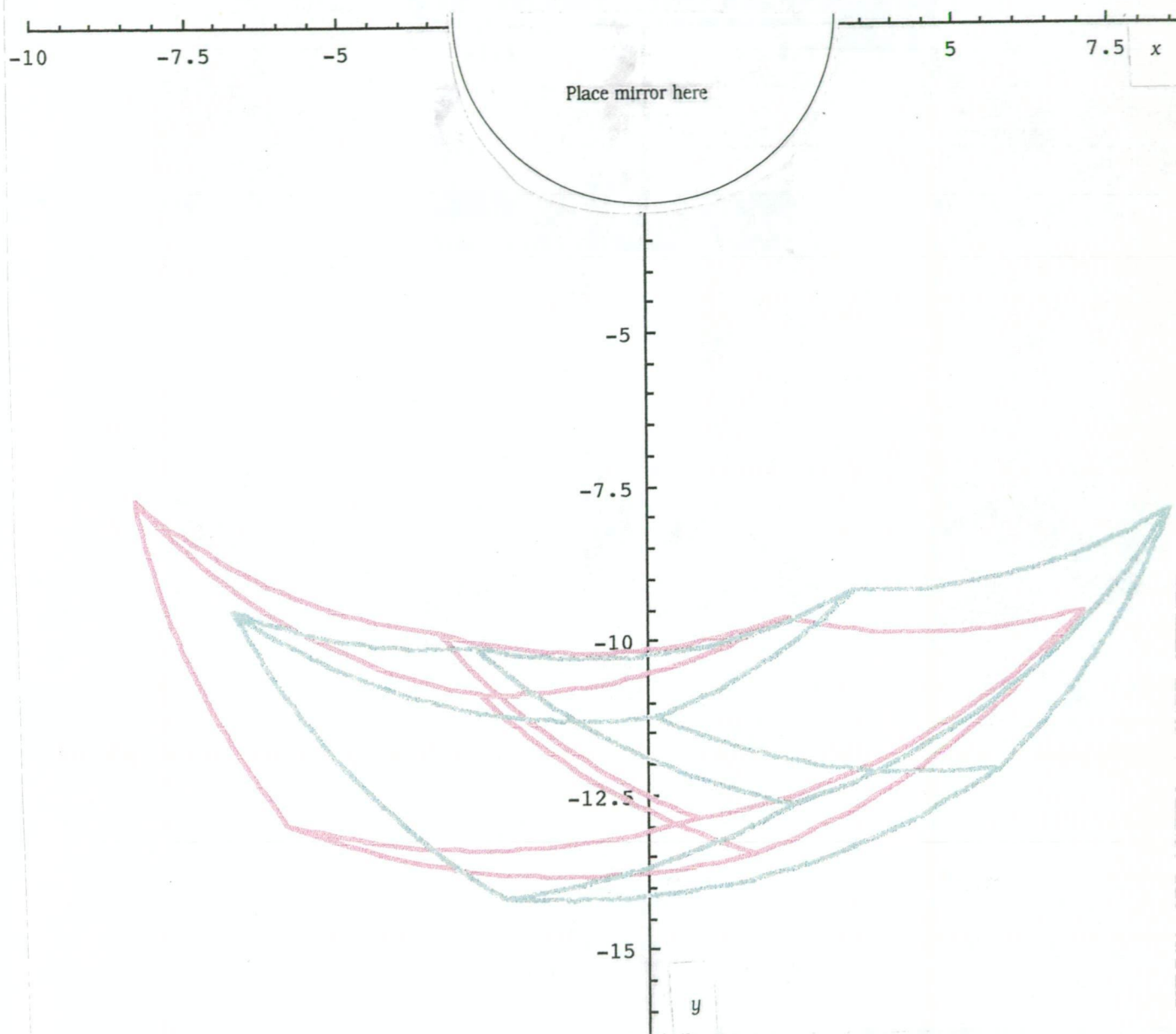


Figure 4.34: Anaglyph for a cylindrical mirror anamorphogram of a cube

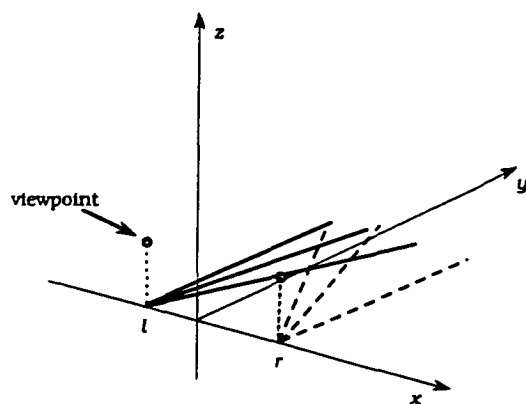


Figure 4.35: Viewpoints directly above l and r

for each eye. We could ask the question:

What is the 'correct' picture that we see from the designated viewpoint in each case?

In our grid example, we saw what appeared to be the grid of Figure 4.4. This 'correct' grid, which the computer drew on the picture plane, $y = 0$, was the perspective drawing of the 'distorted' grid.

Our prescribed viewing of Figure 3.70 gives us a clue as to a possible picture surface for our perspective drawings in this case. We see lines which appear to be parallel, and which appear to intersect our anamorphogram plane (the page) on the Vieth-Muller circle for our fixation point. Consequently, a suitable picture surface may be the inside surface of a cylinder with horizontal cross-section equal to the Vieth-Muller circle for a given fixation point.

To check this supposition we will consider the combination of two such anamorphograms. This allows us to choose a suitable cylinder, since, as we saw in subsection 3.5.2, the intersection points of the sub-pencils of the two anamorphograms define various Vieth-Muller circles.

For convenience, we will choose the points, $l=(-2.875, 0, 0)$ and $r=(2.875, 0, 0)$, to be points on the x -axis of our rectangular cartesian co-ordinate system. The co-ordinates are chosen to match the eye-spacing of the author. Our sub-pencils of lines through each point lie on the xy -plane and the fixed angle between neighbouring lines for each sub-pencil, of our example in the scale drawing of Figure 4.36, is $\pi/8$. After choosing a parametric representation for one ray of our pencils, we construct all other rays by carrying out appropriate rotations and/or translations. This involves multiplying by tranformation matrices of the types introduced in subsection 4.2.3. Two

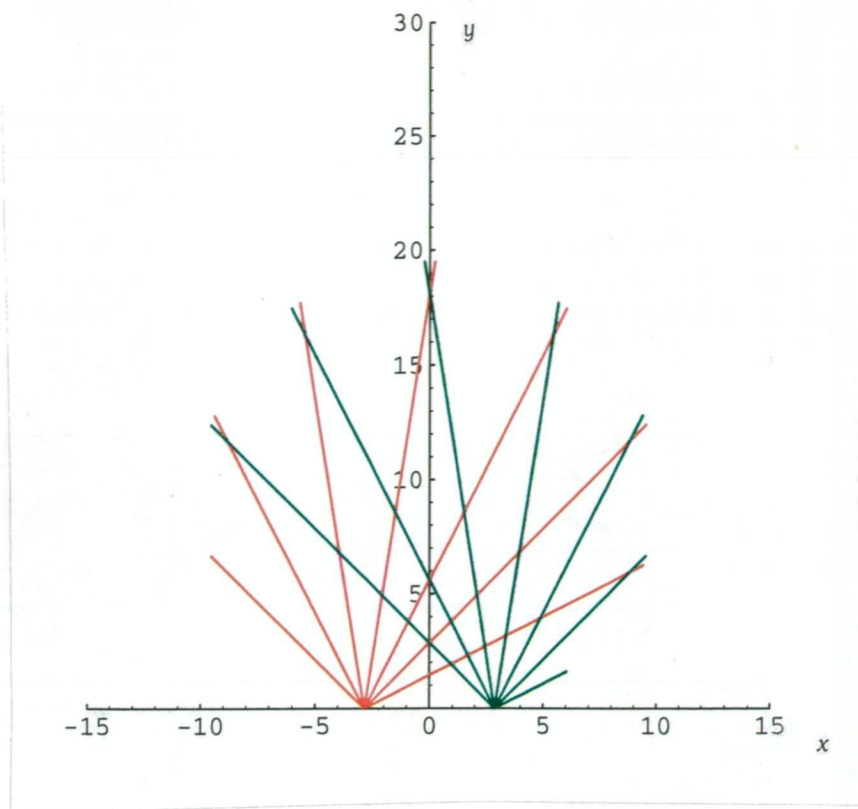


Figure 4.36: Anamorphogram 1...red pencil through l and Anamorphogram 2...green pencil through r

resulting anamorphograms are shown in Figure 4.36. Intersection points such as p_0 , together with the points, l and r , define a circle. Our picture surface will be that of a cylinder through this circle which is perpendicular to our page. We now apply our perspective drawing program of Appendix B to create our perspective drawings; one for each anamorphogram. For the perspective drawings shown in Figure 4.37 (a) and (b), the viewpoint for the red pencil of lines was $(-2.875, 0, 10)$ and for the green pencil, was $(2.875, 0, 10)$. Such viewpoints are directly above l and r , as is required for Luneburg's experiment.

We see that the combined perspective drawing of Figure 4.38 is a very close depiction of what we actually see when we carry out Luneburg's experiment on Figure 3.70. That is, parallel lines intersecting the xy -plane of the original anamorphogram through the intersection points of corresponding rays. It must also be noted that:

1. The perspective drawings of Figures 4.37 and 4.38 are anamorphograms. They must be wrapped inside an appropriate cylinder to be seen as the 'correct' picture.

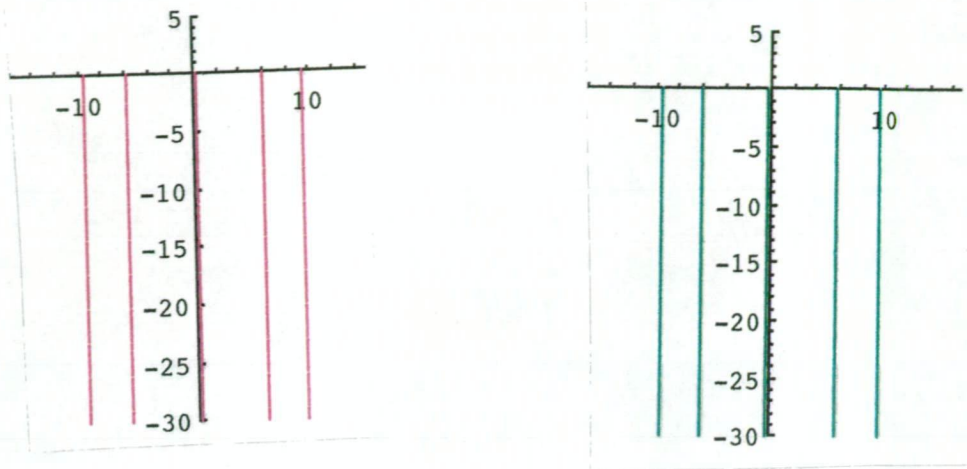


Figure 4.37: Green viewpoint: $(2.875, 0, 10)$; red viewpoint: $(-2.875, 0, 10)$

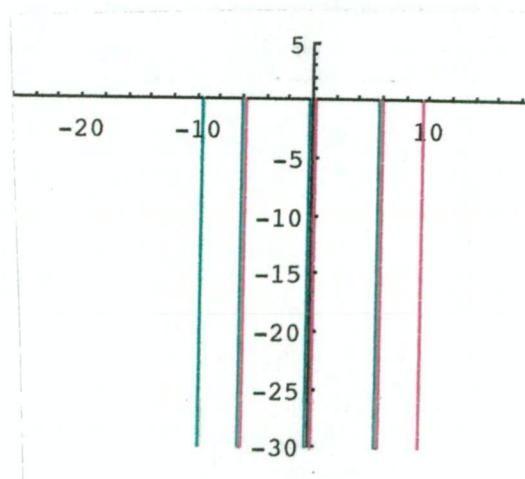


Figure 4.38: Combined perspective drawings for both viewpoints

2. The perspective drawings, for each anamorphogram of Figure 4.36, coincide. Therefore, it seems that to 'fuse' corresponding lines from each sub-pencil, the observer needs to converge the viewlines from each eye on a surface which coincides with the cylinder surface. This contrasts with Luneburg's idea of binocular vision being a projection against a conical surface, however, it does not contradict his theory. As we have seen, it is possible for the eyes and brain to fuse images within small neighbourhoods of geometrically predictable positions: Panum's Fusion region of Figure 3.20 in Chapter 3.

We now devote a chapter to the creation of the more complicated, non-conventional anamorphogram; the Single-Image Stereogram.

Chapter 5

The Construction of Single-Image Stereograms

In Chapter 3 we introduced the notion of a Single-Image Stereogram. Some simple examples such as those in Figures 3.41, 3.42, 3.44, 3.48 and 3.55 were included together with some discussion about how we might view them. There was very little variation in the depth of the perceived pictures in these examples. However, they did give some inkling as to how we may construct a Single-Image Stereogram of other more interesting three-dimensional objects.

5.1 An outline of a construction for Single-Image Stereograms

Let $F(x, y, z) = 0$ be a three-dimensional surface. Then we construct a Single-Image Stereogram of this surface by constructing appropriate horizontal rows of dots on a plane, $y = d$, which is parallel to the xz -plane. In this case, we will choose that the viewer's eyes lie on the x -axis and so, the *distance of the viewer from the Single-Image Stereogram* is d . Each row of dots belongs to a plane, through the viewer's eyes, which cuts the surface. The intersection of this plane with the surface is a plane curve. We construct our stereogram by considering a family of these sloping planes; constructing a row of dots in each. Figure 5.1 illustrates the situation.

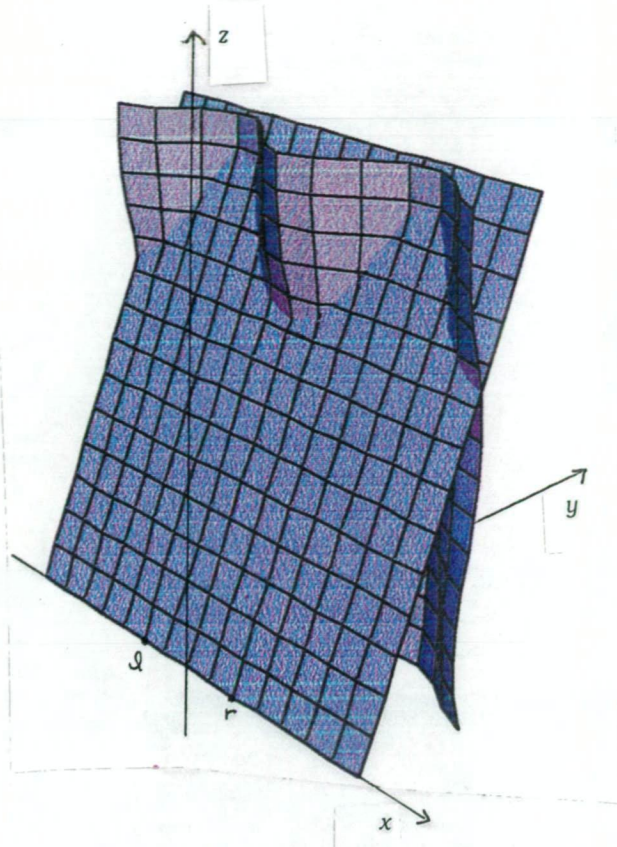


Figure 5.1: Sloping plane through the eyes intersecting a surface

5.1.1 Constructing one row of dots on the xy -plane

We firstly consider the construction of a row of dots on the horizontal plane, $z = 0$, of the above family. In order to find a row of dots which represents the curve $y = f(x)$ for a given domain say, $-q \leq x \leq q$, of Figure 5.2, we carry out the following construction. If the viewer's eyes are situated at the points, l and r , on the x axis of this xy -plane, then to construct a suitable row of dots, along the line $y = d$, we firstly construct the viewline, $l \vee d_1$, from l to a point d_1 on this line. This point, d_1 , represents the position of our first dot and we let D denote the line $y = d$ which lies between the x axis and the curve. Let the line $l \vee d_1$ intersect the curve $y = f(x)$ at a point, p_1 . To find the position of the next dot we draw the viewline $r \vee p_1$, and find its intersection with D . That is,

$$d_2 = (r \vee p_1) \wedge D.$$

To find further dots we repeat this process across the row, next replacing d_1 by d_2 . That is,

$$p_2 = (l \vee d_2) \cap f(x)$$

and so

$$d_3 = (p_2 \vee r) \wedge D.$$

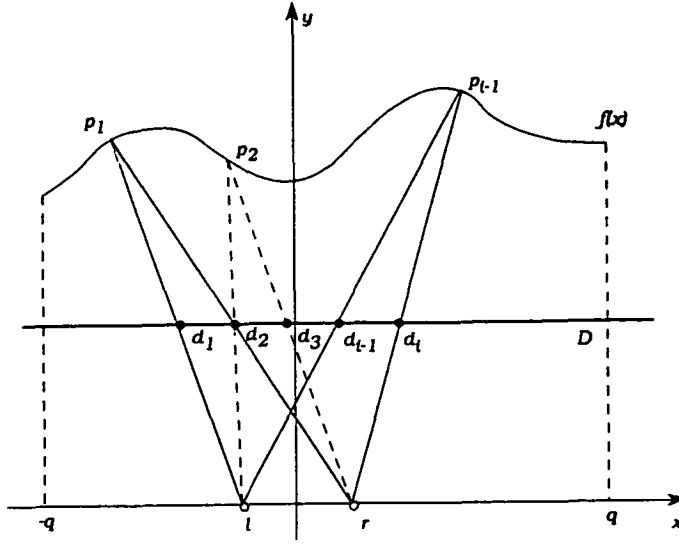


Figure 5.2: Left eye: l ; right eye: r ; viewlines to points, p_i , on curve

In general, the i th dot is given by

$$d_i = (p_{i-1} \vee r) \wedge D, \quad (5.1)$$

$$\text{where } p_{i-1} = (l \vee d_{i-1}) \cap f(x) \mid -q \leq x \leq q : q \in \mathbf{R}.$$

We note that although this procedure means that we are constructing dots from left to right across the row, we could also move to the left across a row using a similar rule

$$d_i = (p_{i-1} \vee l) \wedge D, \quad (5.2)$$

$$\text{where } p_{i-1} = (r \vee d_{i-1}) \cap f(x) \mid -q \leq x \leq q : q \in \mathbf{R}.$$

We use whichever technique is most convenient in any given example. As a result of the possibility of working in either direction, it is not essential to begin at the endpoints of the rows.

We will see some examples later. A computer can easily carry out the iterative process for us.

Various decisions must be made. These are:-

How do we choose our starting point corresponding to p_1 in Figure 5.2, or alternatively, our first dot, such as d_1 ?

In which direction do we move?

How do we know when to stop our iterative process?

Answers to these questions will be discussed for each of the individual examples we consider in this thesis.

We note from Figure 5.2 that a row of dots such as the set of d_i 's on D will leave significant sections of the curve $f(x)$ which are not represented by

dots. To create a stereogram which represents a smooth curve, we need to increase the number of dots. One way of doing this is to consider any *starting interval*, (d_i, d_{i+1}) , of dots $[(d_1, d_2)]$ as shown in Figure 5.2]. If we randomly add dots to this interval, then we can use our iterative process described in equations 5.1 and 5.2 to find the matching dots for these additional dots in adjacent intervals. The process is again illustrated in Figure 5.3. As we saw in Chapter 2, random dots help in the disguising of our picture for monocular viewing.

We note that the number of random dots we add to a starting interval will

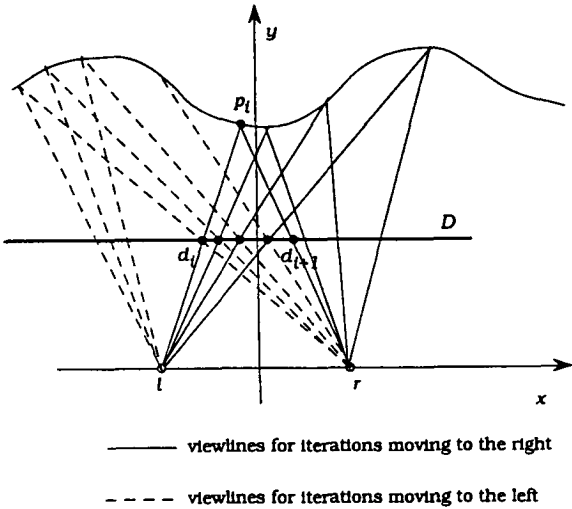


Figure 5.3: Adding random dots to the starting interval

affect the appearance of the resulting Single-Image Stereogram. We don't want situations where dots merge into one another to create solid black regions. The size of our dots, which may be programmed, plays a part here. The smoothest visual results occur for very small dots. This is evident if we compare stereograms of the same objects in Figures 6.38 and 6.39. The dot spacing in the starting interval is also relevant. This is illustrated in the extreme example in Figure 5.4, where the distinct dots in the interval (d_i, d_{i+1}) map onto a solid black line.

Conversely, if our starting interval was as shown in Figure 5.5, then our random choice of dots in this narrower interval could lead to large gaps in another interval. In practice, these gaps may or may not be of consequence. In some cases, though, they mean that some vital sections of the curve would be missed with the viewlines. If there is a significant difference between the

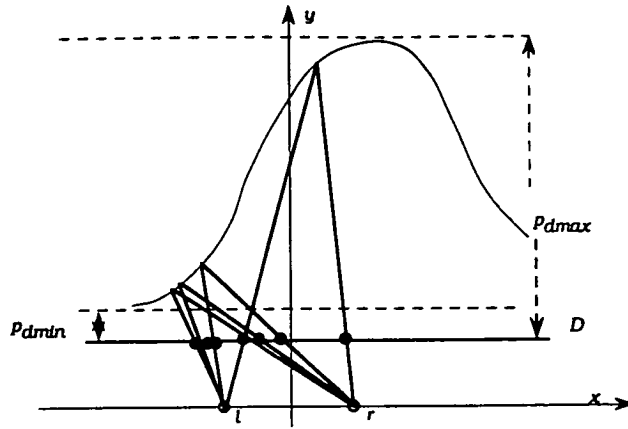


Figure 5.4: Solid black regions are created in the interval to the left

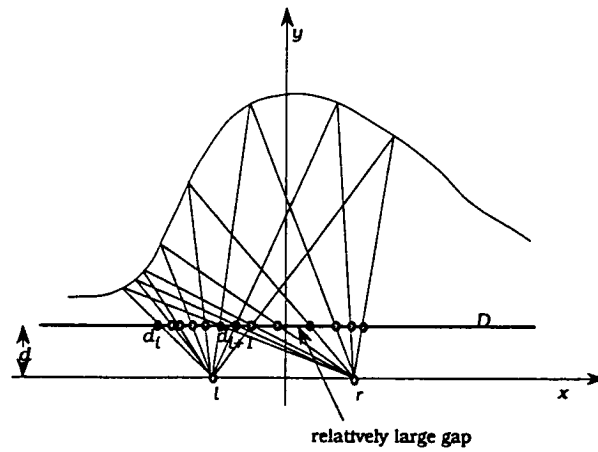


Figure 5.5: Large gaps are created in the interval to the right

maximum and minimum depths, p_{dmax} and p_{dmin} , relative to the distance, d , of the viewer from the stereogram, then care needs to be taken in choosing the starting interval.

5.1.2 Comparisons with some published papers

Terrell and Terrell [27, page 720] comment about this effect of merging dots, however, talk in terms of choosing a ‘strip width’. Maeder’s [17] solution to this problem was to write his programs so that the surface $F(x, y, z) = 0$ lies between two planes, P_1 and P_2 . For our co-ordinate system with the viewer’s eyes on the x -axis and the stereogram on the plane $y = d$, his setup is illustrated for the xy -plane in Figure 5.6. The planes, P_1 and P_2 , are chosen so that they are both parallel to the stereogram, the distance between them is d , which equals the distance of the viewer from the stereogram. The distance between the stereogram and P_2 is d , and the distance between the stereogram and P_1 is $2d$. In Chapter 3, we met the following result

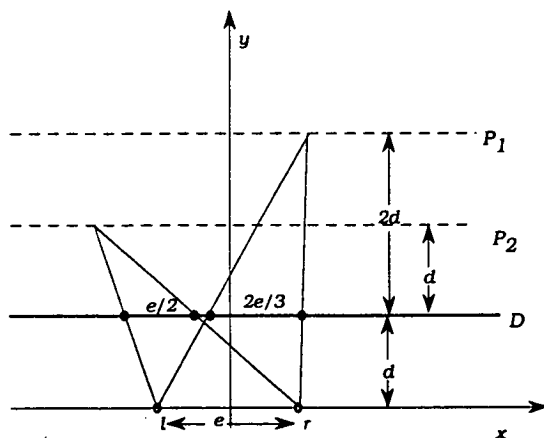


Figure 5.6: Maeder’s depth information lies between the two planes P_1 and P_2 parallel to the plane of the stereogram

$$s/e = p_d/(p_d + d), \quad (5.3)$$

where s is the dot-spacing, e is the viewer’s eye-spacing, p_d is the depth of the perceived point behind the stereogram, and d is the distance of the viewer from the stereogram. Using this result, Maeder’s setup corresponds to a dot spacing between $2e/3$ (for points on P_1) and $e/2$ (for points on P_2). This ensures that either gaps or solid black regions can be avoided, as the differences in dot spacings across the whole stereogram are not very marked. It must also be noted that the largest possible disparity, as measured by the

differences in dot spacing, is $e/6 \approx 0.96\text{cm}$. This also ensures that the dots may be easily fused as, according to our approximate calculations of section 3.4, Chapter 3, their images lie within Panum's fusion region.

These considerations have been included as they are relevant in both the designing and viewing of our Single-Image Stereograms which we introduce later.

It is also appropriate to mention that Terrell and Terrell's [27] computations differ from the ones presented in this thesis in two main respects:

- (i) Their algorithm uses intersections of the surface with horizontal, rather than sloping, planes through the eyes. As they point out, this is a source of error.
- (ii) Their computations avoid computing intersections of lines and surfaces. They make approximations of the value of $f(x)$ for some point, x , in an interval of chosen width. They then use the equation 5.3 to find the associated dot-spacing.

Bar-Natan [1] uses similar approximations to those of Terrell and Terrell while Maeder [17] presents much more accurate results, taking both the sloping planes and the surface-line intersections into account.

Like Maeder's and Bar-Natan's, our programs are written using the programming language *Mathematica*. It must be noted that for solving non-algebraic equations, *Mathematica* uses Newton's iteration method for finding a numerical approximation to the solution. This involves specifying a starting point for the iterations. Hence for surfaces with complicated equations for the curve of intersection of the sloping plane through the eyes, the solutions found by *Mathematica* are approximate.

For the particular cases we will consider in this thesis, this problem does not arise. However, our accuracy does depend on the accuracy of *Mathematica*.

It is also appropriate to consider an extreme case such as that demonstrated in Figure 5.7. We note that the viewline from r to p , cuts the curve, $y = f(x)$, at two points. Physically this amounts to the fact that the right eye would not see the point p . That is, the viewer is too close to the surface to be able to clearly ascertain its shape. This situation is also mentioned on page 44, Chapter 3. If however, this same curve is placed further from the viewer, then for some suitable distance, the viewer would be able to see most points

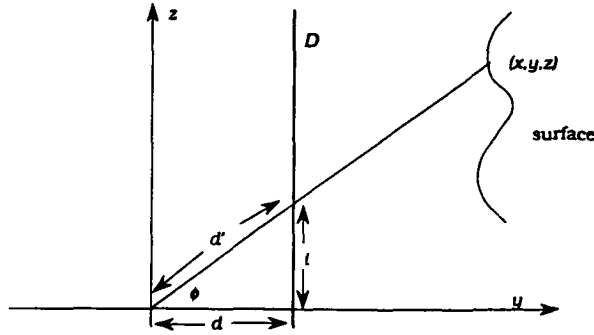


Figure 5.8: Side view of sloping plane through the eyes

the angle of inclination of the sloping plane is given by $\phi = \text{Arctan}[i/d]$. In order to create our row of dots at the height $z = i$ on our stereogram, we need to find the intersection points, (x, y, z) , of this plane with our surface. Once we have found these intersection points, we are effectively working in two dimensions. For convenience, we co-ordinatize each sloping plane relative to the existing x -axis and a y_i -axis, for each i , which is the intersection of this sloping plane with the yz plane. We can think of this as a rotation onto the xy -plane, about the x -axis by an angle of $\text{Arctan}[i/d]$.

Note: The fact that we have specified that the viewer's eyes lie on the x -axis, helps to simplify the calculations here.

This will give us a set of points, $(x', y', 0)$. We then use our algorithm of equation 5.1 to find the positions of the dots on the row of the stereogram which is at a distance of $d' = \sqrt{d^2 + i^2}$ from the viewer. It is convenient to refer to this setup as being on the xy_i plane as shown in Figure 5.9, where the x axis is normal to the page through the origin, o .

Having found a complete row of x co-ordinates of dots where the y_i co-ordinate is $\sqrt{d^2 + i^2}$, we take the x co-ordinates which don't change according to our rotational transformation, and append each of them with our z value, i . This gives a set of ordered pairs which specify the positions of the appropriate dots on our Single-Image Stereogram which is parallel to the xz plane. Specific details of these processes may be found in the individual programs for each specific stereogram. These are included in the appendices.

We will now use our algorithm to create various Single-Image Stereograms. In each individual case, we discuss the special problems involved, and include answers to some of the questions we have raised.

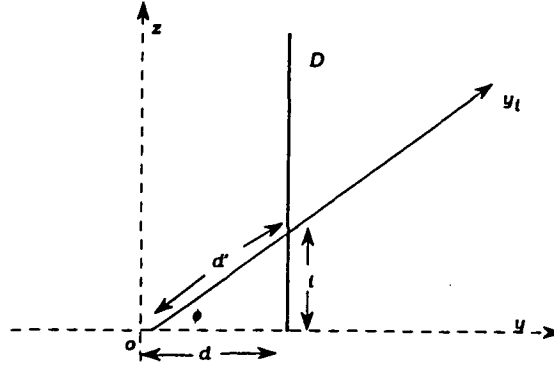


Figure 5.9: Relative positions of the y and y_t axes

5.2 Single-Image Stereogram of an ellipsoid

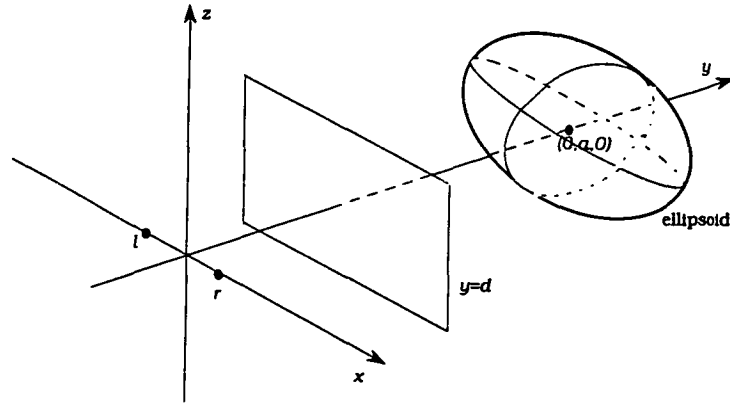


Figure 5.10: Left eye: l , right eye: r ; stereogram at $y = d$; ellipsoid centered at $(0, a, 0)$

We will now consider the problem of creating a Single-Image Stereogram, of a given ellipsoid placed symmetrically about the y axis in our co-ordinate system as shown in Figure 5.10. The equation of the ellipsoid is

$$F(x, y, z) = x^2/b^2 + (y - a)^2/c^2 + z^2/f^2 - 1 = 0, \quad (5.4)$$

where the centre of the ellipsoid is $(0, a, 0)$. There are two points to note:

- (i) When $b=c=f$ we have the special case of a sphere.
- (ii) We must take care in choosing suitable values for the constants b, c, f as we want our ellipsoid to be further away from us than the plane of the stereogram.

There are two possible views we can consider:-

- (i) The outside front surface of the ellipsoid, as shown in Figure 5.11.
- (ii) The inside of part of the back surface of the ellipsoid, as shown in Figure 5.12.

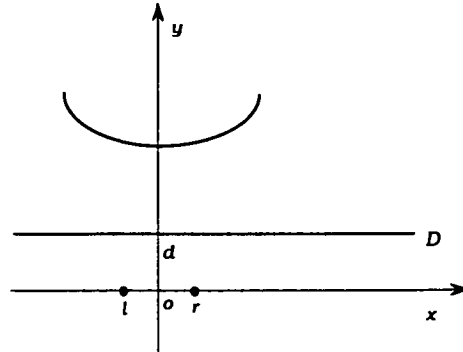


Figure 5.11: Outside front surface of the ellipsoid

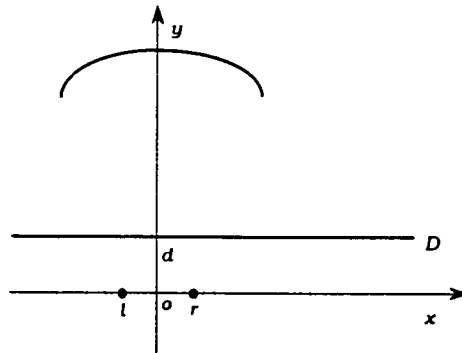


Figure 5.12: Inside back surface of the ellipsoid

Suppose our eyes are placed at positions with co-ordinates $(-w, 0, 0)$ and $(w, 0, 0)$ and that the plane of the stereogram is $y = d$. That is, we are viewing the stereogram from a distance of d cm.

In subsection 5.1.3 we have seen that each row of dots lies on a line with z -co-ordinate, say k , which is the line of intersection of two planes; the plane containing the stereogram, and the plane through our eyes with inclination $\phi = \text{Arctan}[k/d]$ from the horizontal xy plane. In order to construct our stereogram, we need to find the equation of the figure of intersection of this latter plane through our eyes with the ellipsoid. To obtain

the equation of this plane we consider the situation of Figure 5.13 where $\vec{n} = (e, f, g)$, the normal vector, is a vector perpendicular to both the vectors, $\vec{l} = (2w, 0, 0)$ and $\vec{p} = (w, d, k)$, in the plane.

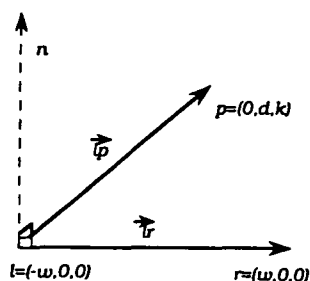


Figure 5.13: l, r, p are planar vectors

Now

$$(2w, 0, 0) \cdot (e, f, g) = 0$$

$$(w, d, k) \cdot (e, f, g) = 0$$

$\Rightarrow e = 0$ and $f = -kg/d$. There are infinitely many vectors perpendicular to the plane but we need only choose one. Choose $g = 1$, then $\vec{n} = (0, -k/d, 1)$ and let $(0, 0, 0)$ be the fixed point in the plane, then the equation of the plane is

$$0x - ky/d + z = 0$$

$$\text{or} \quad -ky/d + z = 0.$$

Consequently, any point on the required figure (in this case, an ellipse) of intersection satisfies both

$$z = ky/d \quad \text{and}$$

$$x^2/b^2 + (y - a)^2/c^2 + z^2/f^2 - 1 = 0.$$

To find the equation of this ellipse, it is appropriate to rotate our axes to new x, y_k, z_k axes, so that the z -term disappears. We can then use our algorithm on our sloping plane which will be the xy_k -plane of our new co-ordinate system.

Let the new co-ordinate axes have the same origin, but be rotated about

the x axis so that the angle between both y and y_k , and z and z_k , is $\theta = \arctan[k/d]$. This is shown in Figure 5.14. If a point, p , has co-ordinates

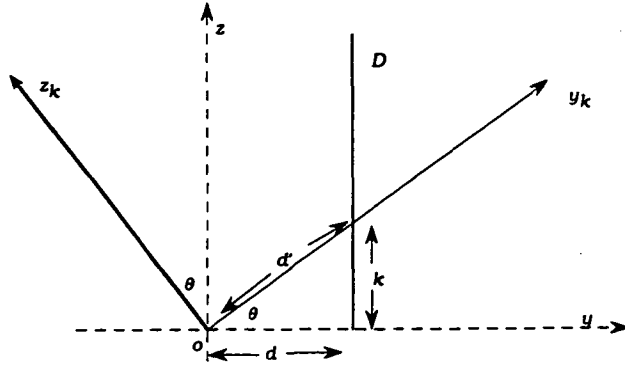


Figure 5.14: Rotation of axes through an angle of $\theta = \arctan[k/d]$

(y, z) with respect to the old axes and (y_k, z_k) with respect to the new axes, then

$$y = y_k \cos \theta - z_k \sin \theta \text{ and}$$

$$z = y_k \sin \theta + z_k \cos \theta.$$

Now $\cos \theta = d/\sqrt{d^2 + k^2}$ and $\sin \theta = k/\sqrt{d^2 + k^2}$. Therefore

$$y = y_k d / \sqrt{d^2 + k^2} - z_k k / \sqrt{d^2 + k^2}$$

and for the ellipse on the xy_k plane we have $z_k = 0$. Hence, if we make the transformation

$$y = dy_k / \sqrt{d^2 + k^2}$$

we find that the equation of the ellipse (for any k) is given by

$$x^2/b^2 K + (y_k - (adf^2 \sqrt{d^2 + k^2} / (f^2 d^2 + c^2 k^2)) / (c^2 f^2 (d^2 + k^2) / (f^2 d^2 + c^2 k^2))) K = 1$$

where

$$K = (f^2 d^2 + c^2 k^2 - a^2 k^2) / (f^2 d^2 + c^2 k^2).$$

Now if we consider the way in which we view an ellipsoid, then Figure 5.15 shows, that in every plane, our right eye sees no point further to the left than the tangent point, t_r , of its viewline and the left eye sees no point further to the right than the tangent point, t_l , of its viewline. However, the right eye will see points of the ellipse further to the right than t_l , and similarly, the left eye will see points further to the left than t_r . Consequently, a suitable starting dot (firstdot in Figure 5.15, and in our program of Appendix E)

from the left for our stereogram is at the intersection point of the viewline from the left eye to the tangent point, t_r , with the line $y_k = \sqrt{d^2 + k^2}$ (in our new co-ordinate system). In order to “see” this tangent point on the ellipsoid, the left eye must view this starting dot when the right eye views the dot, (d_r in Figure 5.15), at the intersection point of its tangent line with the line $y_k = \sqrt{d^2 + k^2}$. Consequently, if any dots are placed in the interval

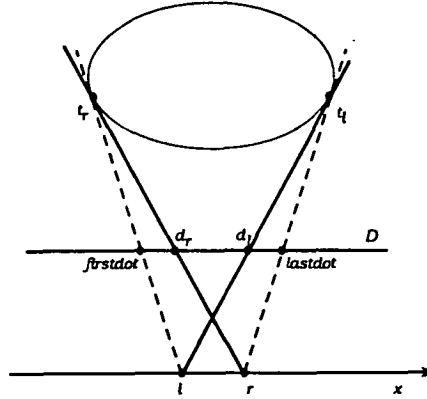


Figure 5.15: First and last dots for each eye

between the firstdot and d_r , then they are used only by the left eye for “viewing” the ellipsoid. We have an analogous situation at the right-hand side of the ellipse, where the interval of dots between d_l and lastdot (see Figure 5.15) is used only by the right eye for “viewing” the ellipse.

Now consider our algorithm for constructing the stereogram which will enable the viewing of the ellipsoid. In any plane, for given k , we start by considering one row of dots. For any dot, d_r , which is viewed by the right eye we can find the next dot required for the right eye by viewing d_r with the left eye. This latter viewline, $l \vee d_r$, has two intersection points on the ellipse. If we are “viewing” the front of the ellipse [illustrated in Figure 5.11] then the next dot for our stereogram is given by $d_{r+1} = (i_1 \vee r) \wedge D$ and if we want to “see” the inside back [illustrated in Figure 5.12] of the ellipse then the next dot to the right is given by $d'_{r+1} = (i_2 \vee r) \wedge D$.

For a complete row of dots we may iteratively apply this process to an interval of dots to the far left (for convenience). The number of dots randomly chosen in this interval depends on the size and required denseness of the stereogram. The x co-ordinates of this interval of dots are randomly selected between two given values. The iterations finish for each row when the dot obtained is further to the right than the last dot.

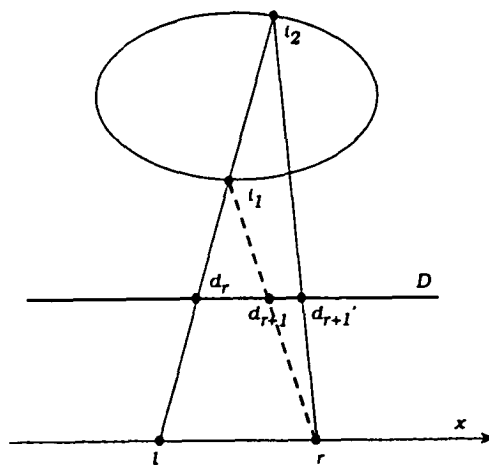


Figure 5.16: Matching dots and intersections with the ellipse of viewlines

To complete our full Single-Image Stereogram we repeat the above process for a selection of possible values for k as described in subsection 5.1.3. For the ellipsoid, k will vary between the z co-ordinates of the tangent points of the planes through our eyes, as shown in Figure 5.17. We simply find the

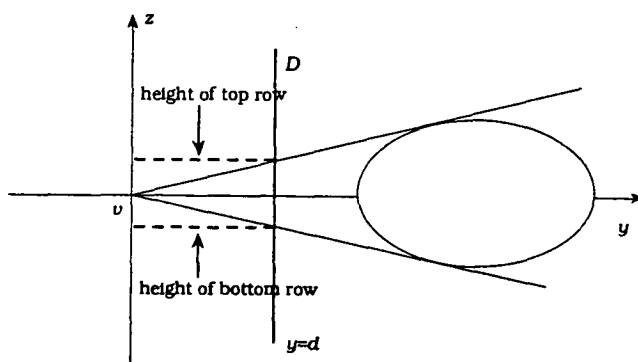


Figure 5.17: Tangent points on ellipsoid of plane through our eyes.

value of k for which the major and minor axes, of the ellipse of intersection, have length 0.

5.2.1 Boundary considerations and further questions.

If our starting interval to the left includes the first dot, then we will perceive a boundary point of the ellipse on the left, however, by iterating to the right until the x co-ordinate of a dot exceeds a set value, we have very little chance of “seeing” a point on the right-hand boundary. Hence, the program is written so that it runs alternately, from left to right (giving a point on the

left boundary), or from right to left (giving a point on the right boundary).

The Single-Image Stereogram obtained by just considering an ellipsoid without any background, or surroundings, is unsatisfactory for viewing because on either side, we have an interval of dots which is viewed by one eye only. This means that these intervals are seen as flat regions on the page around the ellipsoid which detracts from the three-dimensional effect. This can be seen by viewing Figure 5.18.

To remedy this situation, a background or surrounding plane was intro-

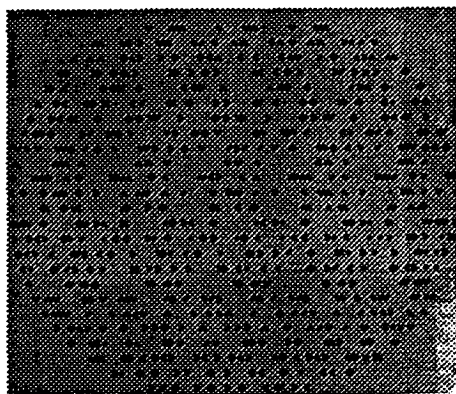


Figure 5.18: Single-Image Stereogram of the inside of a sphere with no background.

duced so that both the boundary intervals of dots were viewed by the opposite eye for viewing the background. Two intervals of dots viewed by one eye only still exist to the far left and right of our stereogram, however, in this instance, do not detract from the viewing of our central surface, the ellipsoid.

The introduction of the background plane causes new questions to arise:

1. Is it best to include the points on the boundary or is a vague edge better?
2. Where should we place the plane in relation to the ellipsoid in order to obtain a satisfactory Single-Image Stereogram?
3. What is a satisfactory Single-Image Stereogram?

Other questions we may ask include:

4. When do inaccuracies, such as not considering sloping planes through the eyes, become obvious when viewing?
5. What happens if the tangents intersect behind the Single-Image Stereogram?

The answers to such questions are largely a matter of opinion, however we note that the answer to the last question lies in an examination of Figure 5.19. We see that there is no overlap, on D , of the possible sets of dots for

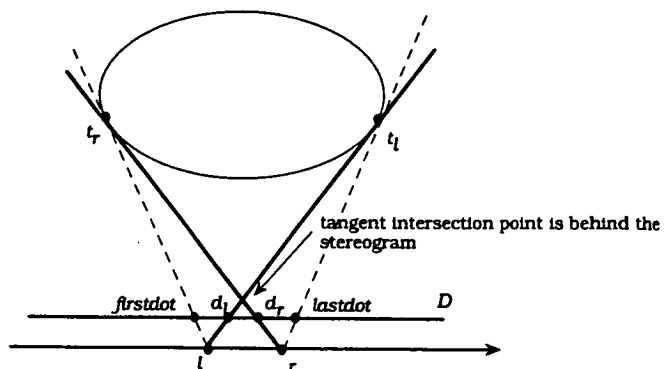


Figure 5.19: Tangents intersect behind the stereogram.

each eye. Consequently, for such a special case, we could only represent this ellipsoid by a two-picture stereogram. Provided that this situation occurs for such tangents to the ellipsoid on every sloping plane through the eyes.

The associated program in *Mathematica* is explained in Appendix E. Some resulting Single-Image Stereograms are shown in Figures 5.20, 5.21, 5.22 and 5.23.

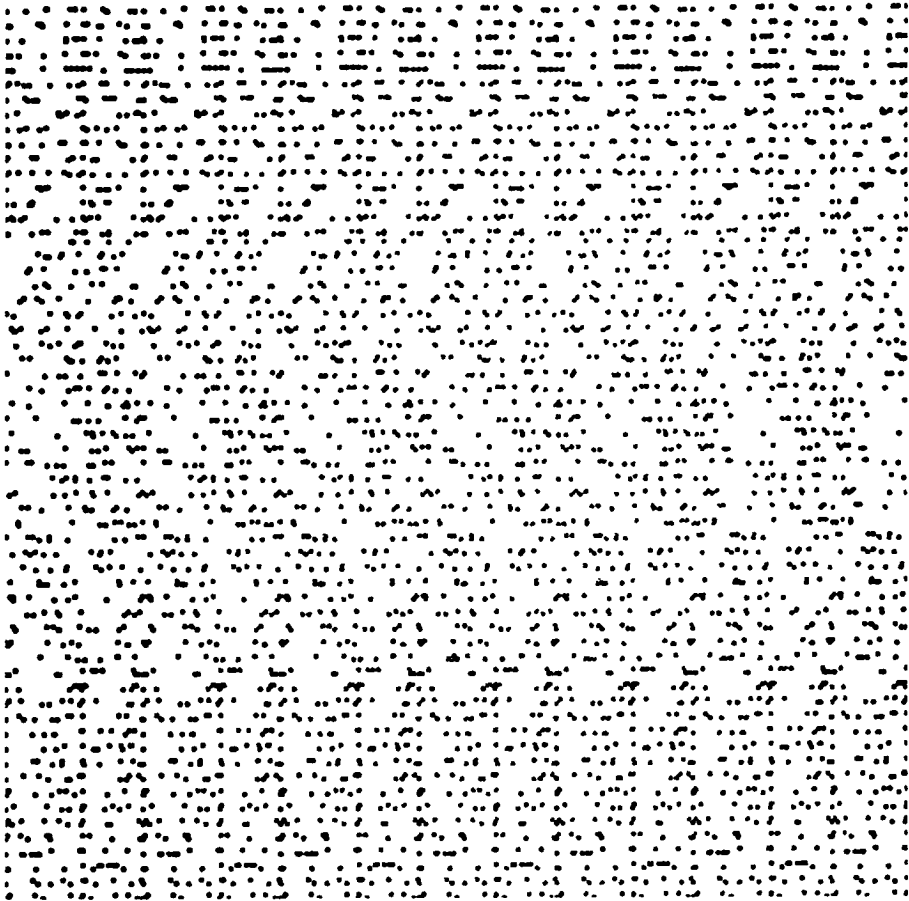


Figure 5.20: Single-Image Stereogram of the inside of a sphere.

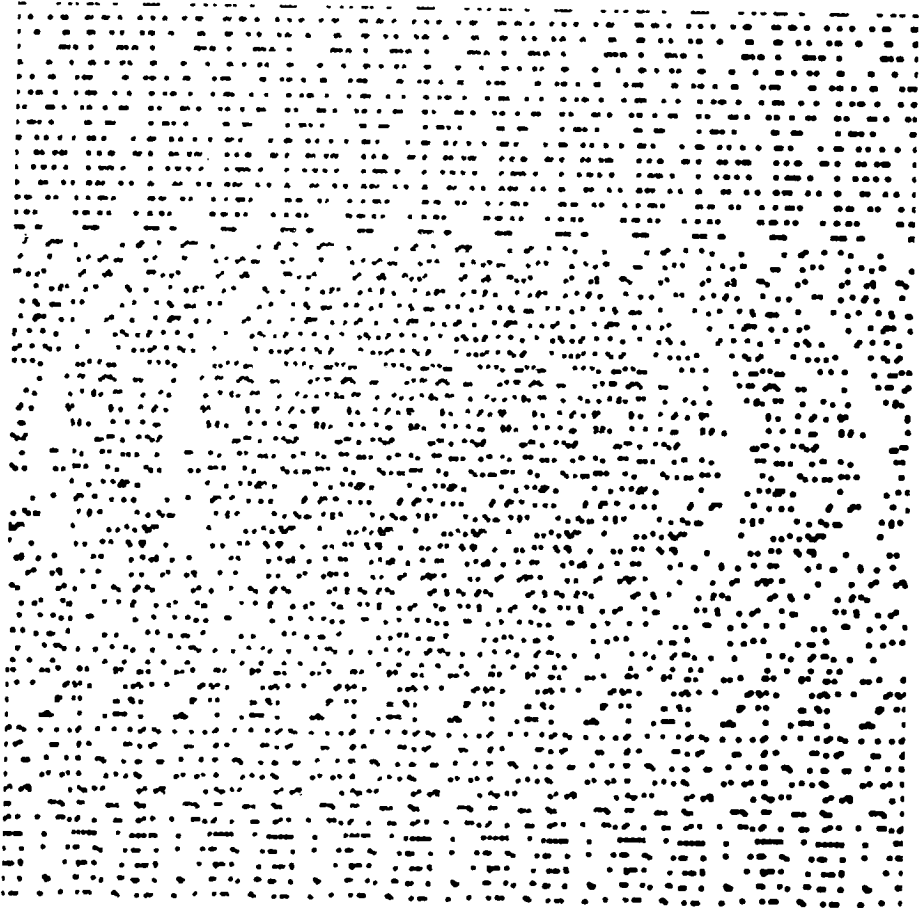


Figure 5.21: Single-Image Stereogram of the outside of a sphere.



Figure 5.22: Single-Image Stereogram of the inside of an ellipsoid.



Figure 5.23: Single-Image Stereogram of the outside of an ellipsoid.

5.3 Single-Image Stereogram of cube

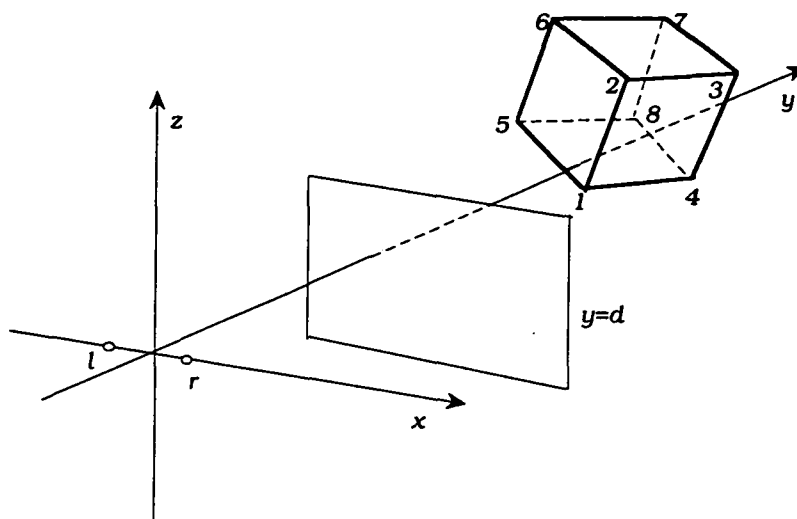


Figure 5.24: Left eye: l , right eye: r ; stereogram at $y = d$; cube with vertices 1,2,3,4,5,6,7,8

We now consider the problem of creating a Single-Image Stereogram which allows us to see a three-dimensional cube, as distinct from the perspective drawing of one on a two-dimensional page. If we view the anaglyph of Figure 4.32, then we see the boundary line segments, or edges, of such a cube. Its Single-Image Stereogram would enable us to 'see' some of its faces as well. The creation of such a Single-Image Stereogram is considerably more complicated than our preceding ellipsoid example of section 5.2. If we consider a sloping plane, through the nodal points of our eyes, which intersects the cube, then the specification of the figure of intersection on this plane, is not as easy to obtain as the ellipse was, in the case of the ellipsoid. We must take into account the fact that, when we view a cube, the number of faces visible to each eye, may be one, two or three.

5.3.1 Which faces of the cube can we see?

Lemma 5.1 *Consider a plane containing a face of a cube. This face is visible if and only if the viewpoint, v , is on the opposite side of the plane to the cube (see Figure 5.25).*

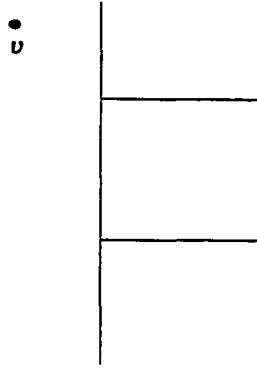


Figure 5.25: View looking down on cube, plane and viewpoint.

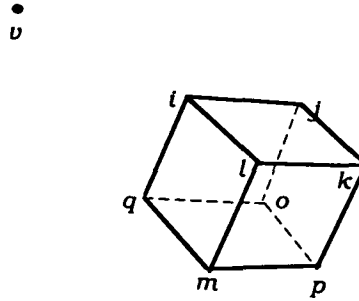


Figure 5.26: Cube and viewpoint

Lemma 5.2 : *There exist three faces of a cube which are never visible from a viewpoint v .*

Proof: Suppose the face $iqml$, of Figure 5.26, is visible, then v and the cube are on the same side of the plane, $o \vee p \vee k$. This is the plane through the opposite, and parallel, face $opkj$ of the cube. Hence this opposite face is not visible by lemma 5.1. Since there are 3 pairs of such opposite and parallel faces on a cube, we know that at least 3 faces are not visible. \diamond

Lemma 5.3 *Consider the cube of Figure 5.26. Let α_x denote the angle between the vectors $\underline{v} - \underline{i}$ and $\underline{x} - \underline{i}$, where i represents the closest vertex to the viewpoint v , $0 \leq \alpha_x \leq \pi$ and x is any vertex adjacent to i . (see Figure 5.27)*

The face ijl is visible $\Leftrightarrow \alpha_q > \pi/2$

$$\Leftrightarrow (\underline{v} - \underline{i}) \circ (\underline{q} - \underline{i}) < 0$$

Proof: Face ijl is visible \Leftrightarrow cube is on the opposite side of the plane $i \vee j \vee l$ from v . (from Lemma 5.1)

$$\Leftrightarrow \alpha_q > \pi/2 \quad (\underline{q} - \underline{i} \perp i \vee j \vee l)$$

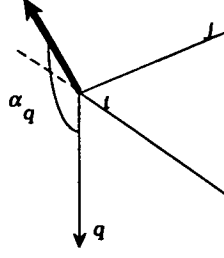


Figure 5.27: α_q is the angle between vectors $\underline{v} - \underline{i}$ and $\underline{q} - \underline{i}$

$$\Leftrightarrow \cos \alpha_q < 0$$

$$\Leftrightarrow ((\underline{v} - \underline{i}) \circ (\underline{q} - \underline{i})) / |\underline{v} - \underline{i}| |\underline{q} - \underline{i}| < 0$$

$$\Leftrightarrow (\underline{v} - \underline{i}) \circ (\underline{q} - \underline{i}) < 0.$$

◇

If we know the co-ordinates of the eight vertices of the cube and the viewpoint, then using the result of Lemma 5.3 we can establish which faces are visible in terms of their vertices. We have three possible cases: one, two or three faces are visible.

For example,

- (i) $\alpha_q > \pi/2$; $\alpha_j, \alpha_l < \pi/2 \Rightarrow$ one face is visible. (That is, the face $ijkl$.)
- (ii) $\alpha_j, \alpha_q > \pi/2$; $\alpha_l < \pi/2 \Rightarrow$ two faces are visible. (That is, faces $ilmq$ and $ijkl$.)
- (iii) $\alpha_j, \alpha_l, \alpha_q > \pi/2 \Rightarrow$ three faces are visible. (That is, faces $ilmq$, $ijkl$ and $ijom$.)

We have already discussed a method for finding the eight vertices of a cube in section 4.2.3. We will again use this technique. For the purposes of finding the number of visible faces, we will take the viewpoint to be at the origin, which represents the position of the *midpoint of the viewer's eyes*. That is, the midpoint of the line segment joining the nodal points of the viewer's eyes. Our function for finding the number of visible faces is called **nvisiblefaces** in our program of Appendix F. It must be noted that this latter viewpoint is different from the respective viewpoints, at l and r , which are vital for the construction of our Single-Image Stereogram.

We will now discuss the method of construction of our Single-Image Stereogram in detail. In order to 'see' our cube we will need to use our "uncrossed" viewing technique of subsection 3.2.1. Our description needs to be broken into several parts.

5.3.2 Single-Image Stereogram representing a quadrilateral on one plane in space

Firstly, we consider a Single-Image Stereogram which will allow us to see part of a plane, Π . Each of the visible faces of a cube, or in general a rectangular prism, will be quadrilaterals. Consequently, the part of the plane we consider will be a quadrilateral, Π' , with vertices $(\alpha_1, \beta_1, \gamma_1)$, $(\alpha_2, \beta_2, \gamma_2)$, $(\alpha_3, \beta_3, \gamma_3)$ and $(\alpha_4, \beta_4, \gamma_4)$.

Our co-ordinate system is as shown in Figure 5.24. We again consider our sloping plane, with equation $z = ky/d$, through our eyes which intersects the plane, Π , of our stereogram at a height of $z = k$. This plane which we

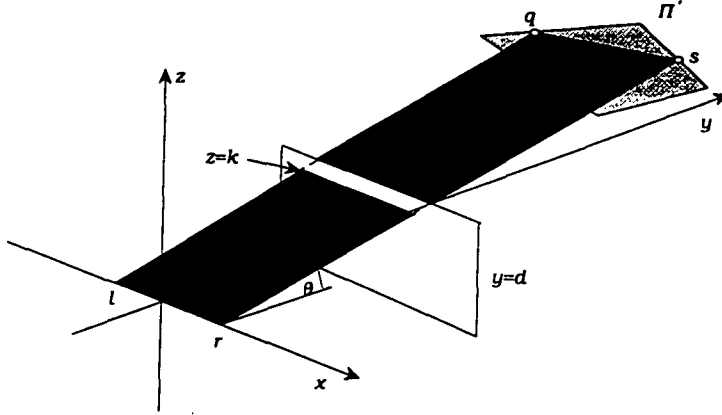


Figure 5.28: Sloping plane, Θ , through the eyes cuts the quadrilateral, Π'

have labelled Θ in Figure 5.28, will intersect the plane Π in a line, L , and it is a particular segment of L that we are interested in. That is, the segment on the quadrilateral Π' which is \overline{qs} in Figure 5.29.

Consider firstly, the endpoints q and s of this segment. To find their co-ordinates, we consider the parametric representations of each of the four lines containing the sides of the quadrilateral Π' . The line segment with endpoints $(\alpha_i, \beta_i, \gamma_i)$ and $(\alpha_{i+1}, \beta_{i+1}, \gamma_{i+1})$, may be represented by

$$x = \alpha_i + (\alpha_{i+1} - \alpha_i)t,$$

$$y = \beta_i + (\beta_{i+1} - \beta_i)t,$$

$$z = \gamma_i + (\gamma_{i+1} - \gamma_i)t,$$

$$\text{where } t \in \mathbb{R}, \quad 0 \leq t \leq 1.$$

We can now find the co-ordinates of the intersection points q and s by considering the value of the parameter t for each boundary line intersection.

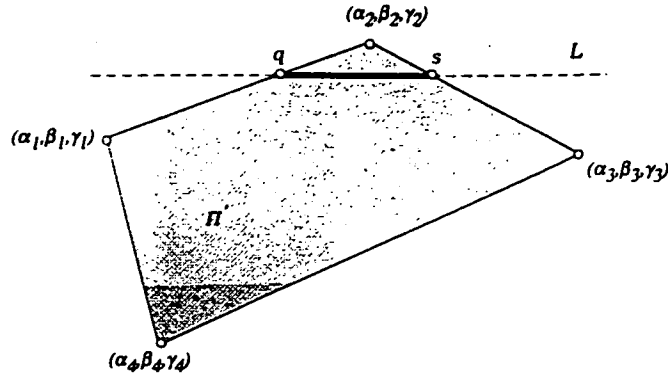


Figure 5.29: Line segment which is the intersection of the sloping plane with Π'

If $0 \leq t \leq 1$ then this is one of our required points.

Note: The plane Θ will intersect every line of the boundary, but not every line segment which is a side of our quadrilateral Π' . By *line of the boundary* we mean a line which contains the line segment which is an edge of the quadrilateral.

5.3.3 Rotation of line segment onto xy -plane

Normally, there will be two intersection points as shown in Figure 5.29, however, there is an exception. This occurs when the plane Θ cuts through a single vertex of Π' . This is a minor consideration when looking at the whole quadrilateral as this one point can be represented on our Single-Image Stereogram by two points only.

Having found two intersection points, q and s , with co-ordinates (x_1, y_1, z_1) and (x_2, y_2, z_2) respectively, we consider the problem of finding the equation of the line containing the segment \overline{qs} . For ease of calculation we will rotate the endpoints q and s so that they lie in the xy -plane. The plane Θ is inclined at an angle $\theta = \arctan[k/d]$ (for each k) to the xy -plane, hence we need to rotate points q and s about the x -axis in a clockwise direction by an angle of θ degrees. To achieve this rotation we multiply each point by the rotation matrix $\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$.

This will give two new points, $q' = (x'_1, y'_1, 0)$ and $s' = (x'_2, y'_2, 0)$ and it is now relatively easy to find the equation of the line through the line segment $\overline{q's'}$. We must also use the fact that after rotation onto the xy -plane,

the row of dots which corresponds to this line segment will be at a distance $d' = (k^2 + d^2)^{1/2}$ from the viewer.

5.3.4 Algorithm for one row of dots/line segment

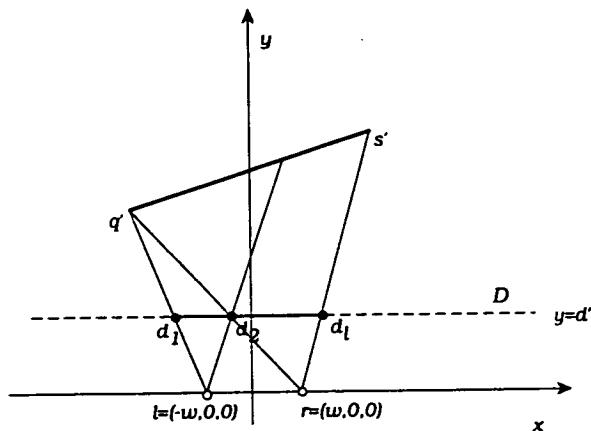


Figure 5.30: Using our algorithm to find the dots

Once we have the equation representing the line through the line segment, we apply our algorithm of equation 5.1. As shown in Figure 5.30, our first dot, d_1 , will be the intersection point of the line $l \vee q'$ with the line $y = d'$ and our last dot, d_l , must be no further to the right than the intersection point of $r \vee s'$ with $y = d'$. If we move from right to left, then the roles of these dots would be reversed.

Having established how to find the dots for a complete row, we now look at the problem of calculating the dots for a selection of possible rows. That is, for possible values, $z = k$, on our stereogram plane. Hence we consider:-

- (i) The minimum and maximum possible z values (in the plane $y = d$), call them z_{min} and z_{max} respectively.
- (ii) The step-size as we move from z_{min} to z_{max} . The number of rows of the stereogram will depend on the step-size which in turn is dependent on the Dot-size. By Dot here, we mean the dot, figure or motif, which we ask the computer to plot at the positions of the calculated dots for the stereogram. Hence, the height and width of such Dots must be considered, so that they do not overlap, or merge into one another. This is a programming problem.

5.3.5 z_{min} and z_{max}

Suppose we want the parametric representation of the lines, L_i , through $(0,0,0)$ and $(\alpha_i, \beta_i, \gamma_i)$ where $i = 1, 2, 3, 4$. This is given by

$$(0,0,0) + t(\alpha_i, \beta_i, \gamma_i), \quad 0 \leq t \leq 1.$$

The origin, $(0,0,0)$, represents the midpoint of the line segment joining the viewer's eyes. We can see from Figure 5.31 that z_{min} for our stereogram,

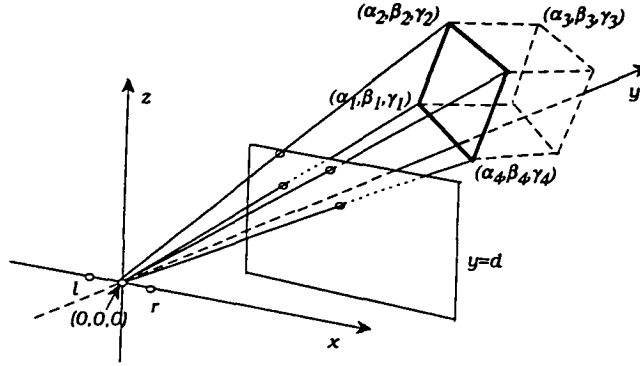


Figure 5.31: Finding maximum and minimum z using viewlines from midpoint of the eyes

corresponding to one face of our cube, will occur at the minimum z value obtained from the four intersection points of L_1, L_2, L_3, L_4 with the plane $y = d$. That is,

$$t(\alpha_i, \beta_i, \gamma_i) = (t\alpha_i, t\beta_i, t\gamma_i) = (t\alpha_i, d, t\gamma_i)$$

$$\Rightarrow t = d/\beta_i, \quad i = 1, 2, 3, 4$$

Therefore

$$z_{min} = \text{Minimum}[\gamma_1(d/\beta_1), \gamma_2(d/\beta_2), \gamma_3(d/\beta_3), \gamma_4(d/\beta_4)].$$

Similarly,

$$z_{max} = \text{Maximum}[\gamma_1(d/\beta_1), \gamma_2(d/\beta_2), \gamma_3(d/\beta_3), \gamma_4(d/\beta_4)].$$

It must be noted that by considering the lines from the midpoint of the eyes, we are making an approximation.

5.3.6 Summary of steps for creating the entire Single-Image Stereogram of cube

We now extend our construction of a stereogram for one face of our cube, to more than one face. The number of faces we consider will depend on how many are visible. This will vary according to the position of the cube with respect to the viewer. The program for creating our Single-Image Stereogram is given, and explained in fine detail, in Appendix F. We have again included a background plane to enhance the three-dimensional effect, and to eliminate the problem discussed in section 5.2.1.

Basically we have the following steps:- We begin with the cube, of dimension $2dim$, in general position about the origin, as shown in Figure 5.32. In order to find the vertices of cubes in varying positions, we carry out

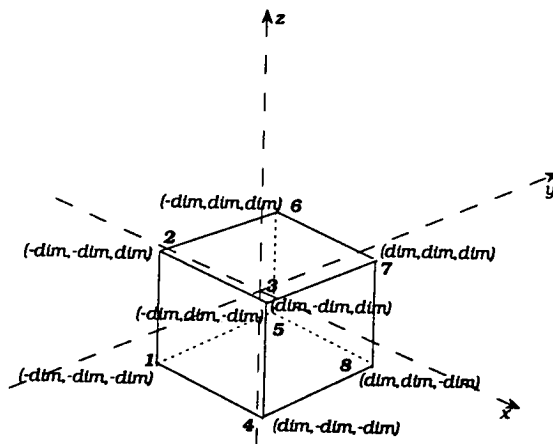


Figure 5.32: Cube of dimension $2dim$, centered at the origin

transformations, rotational and translational (as discussed in section 4.2.3), so that our cube is moved further from the viewer than the plane of the intended stereogram. This plane is again $y = d$, where d is the distance, in centimetres, of the viewer from the stereogram. We now have eight new vertices which will help us specify the faces of the cube which are visible to the viewer. The *closest vertex* to the viewer is defined to be the one of minimum distance from the midpoint of the line segment joining the nodal points of the eyes. In our case, this midpoint is the origin, $(0, 0, 0)$. Having found the closest vertex, we use the result of Lemma 5.3 to find the number of visible faces. Once we have the number of visible faces, and the closest vertex, we can establish the vertices of each visible face. This gives a rather complicated rule which we have called *visible faces* in our program of Ap-

pendix F. Fortunately, the computer does the work for us in applying this rule.

The next step is to use our method of subsection 5.3.5 to find the range of z values, on our stereogram plane, which covers the cube. For our complete stereogram, we construct rows of dots which correspond to values just outside this range of values, but which allow us to, perceive our background plane. This gives greater clarity, of the edge boundaries of our cube, for viewing.

As we move up our rows of dots from the bottom of our stereogram, our dots in each row represent points on different planes. For example, the first row represents points on one plane, the background plane. Then as we move up to our cube, we may have the situation where as we move from left to right across a row of dots we have points represented for four different planes, as shown in Figure 5.33. That is, to construct a row of dots, consideration has

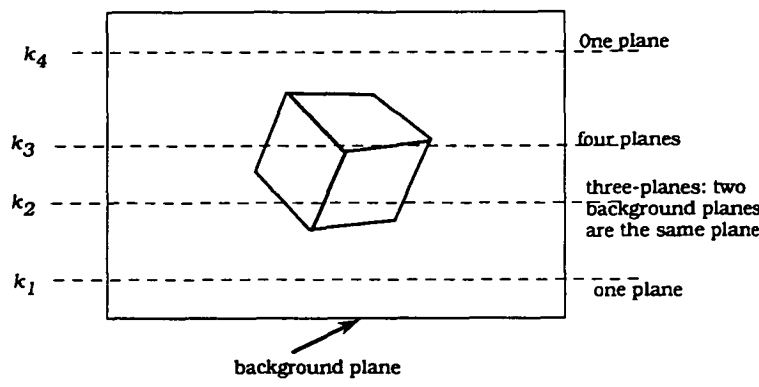


Figure 5.33: Five possible k values for rows of dots representing different planes as we move across a row; three visible faces of the cube

to be given to the problem of working out, for each value of k , which of the visible faces requires dots at that level, and in what order. It is possible to have rows of dots which traverse no faces, one face, two faces or three faces; even when three faces are visible. In our program, the computer function we have named *cuberowdots* solves this problem. To understand how *cuberowdots* works, we will consider one case.

Suppose that there are three visible faces, and that k lies between the minimum and maximum z values for the visible faces, say *visibleface1* and *visibleface3*. This means that this row of dots only crosses two of the faces. If we start at the left-hand side of the cube, then we need to find which

of these two faces is furthestest to the left. This is done by comparing the *firstdot* for visible faces 1 and 3. The function **Firstdot**, of our program, gives us an x co-ordinate, and so the minimum value tells us which face we consider first. The *lastdot* for the first face is used as the first dot for the next face. The last dot for this second face is then used as the first dot for the background plane. Currently, the background plane to the left of the cube is considered separately. This has the disadvantage of creating a stereogram with monocular clues across its central band. Some thought may be given to amending this later.

We now include examples, in Figures 5.34, 5.35 and 5.36, of the stereograms which can be created using our program of Appendix F.

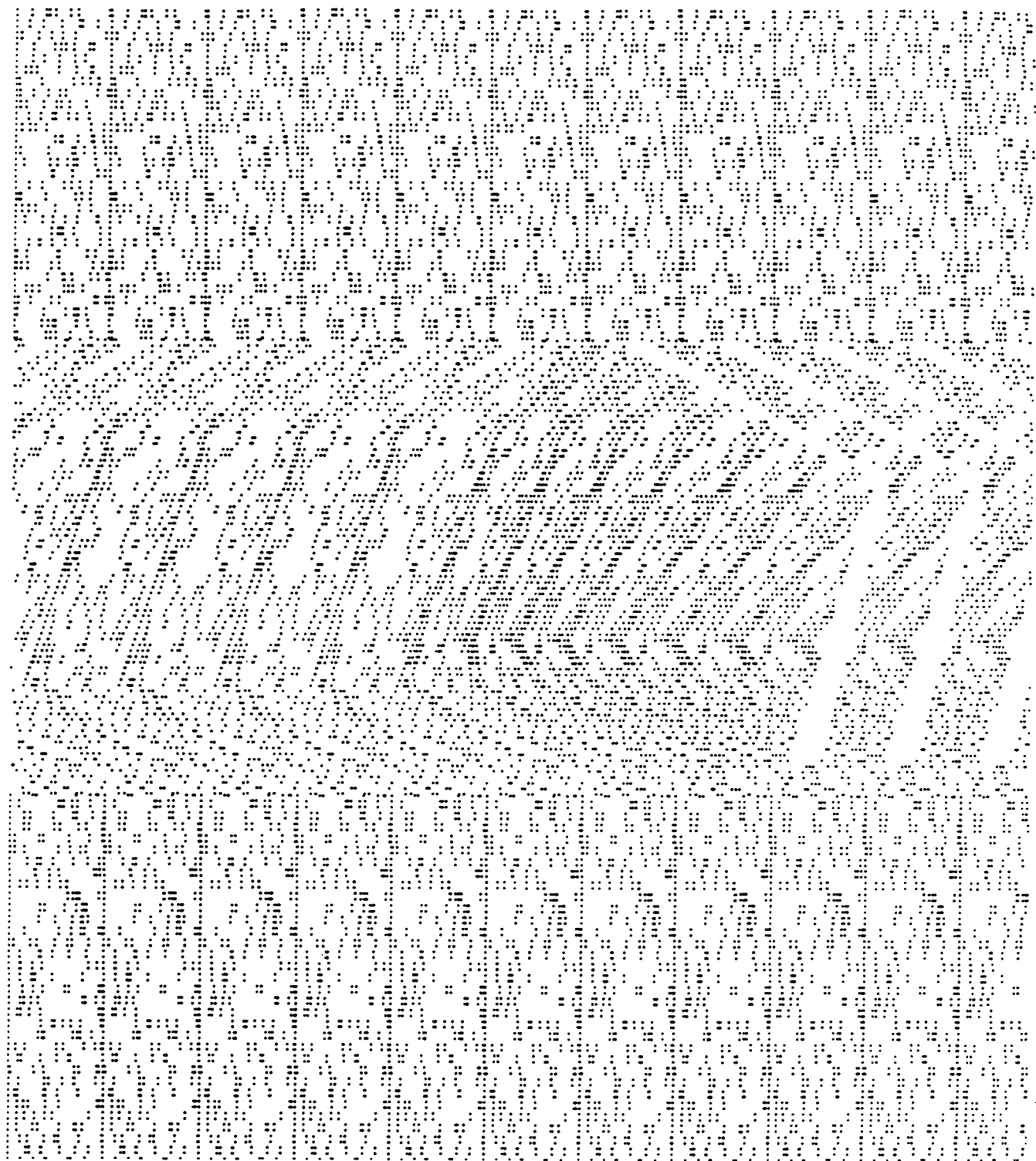
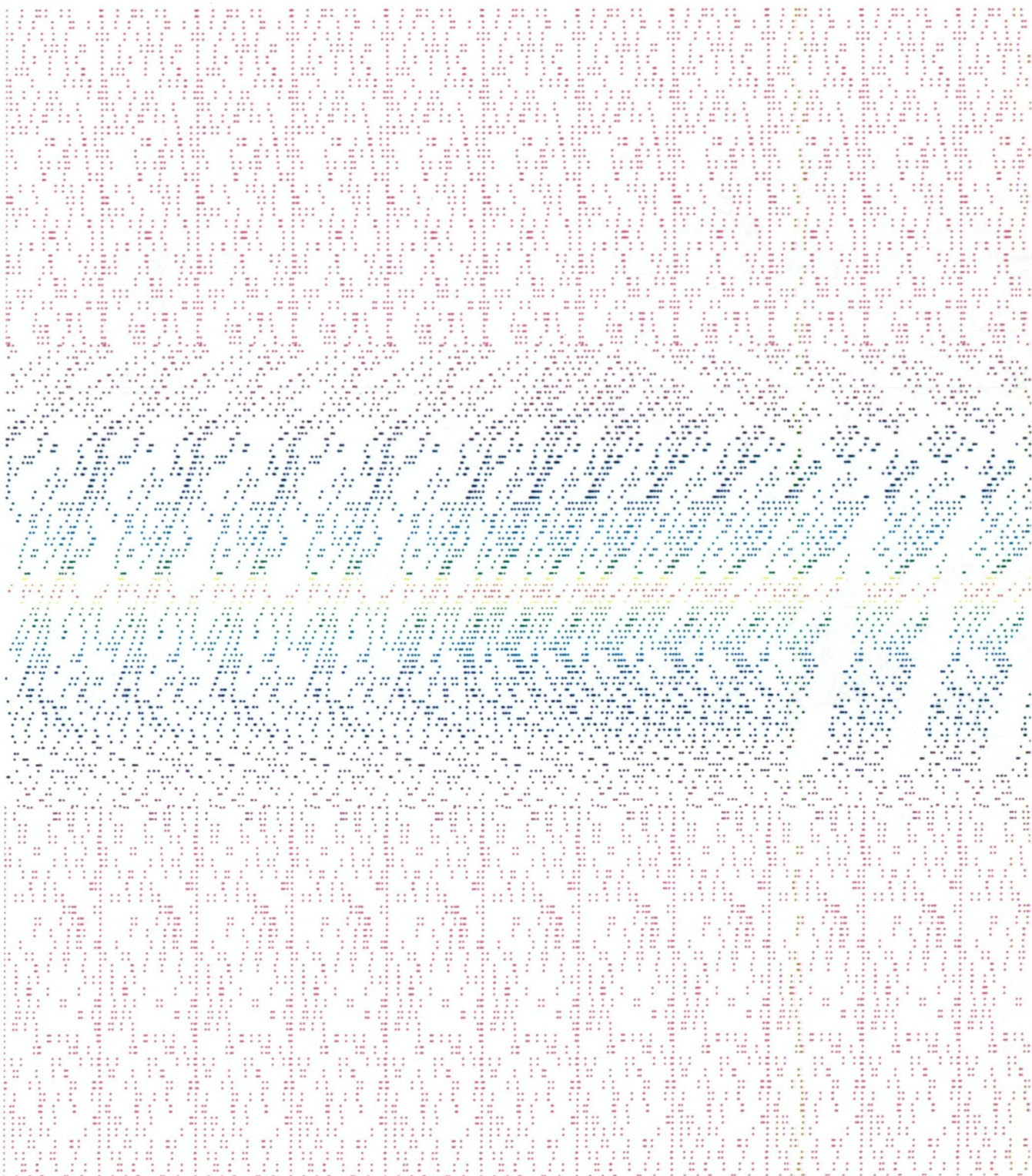


Figure 5.34: Three visible faces of a cube; viewpoint: $(0, -30, 0)$; origin is at the centre of the page



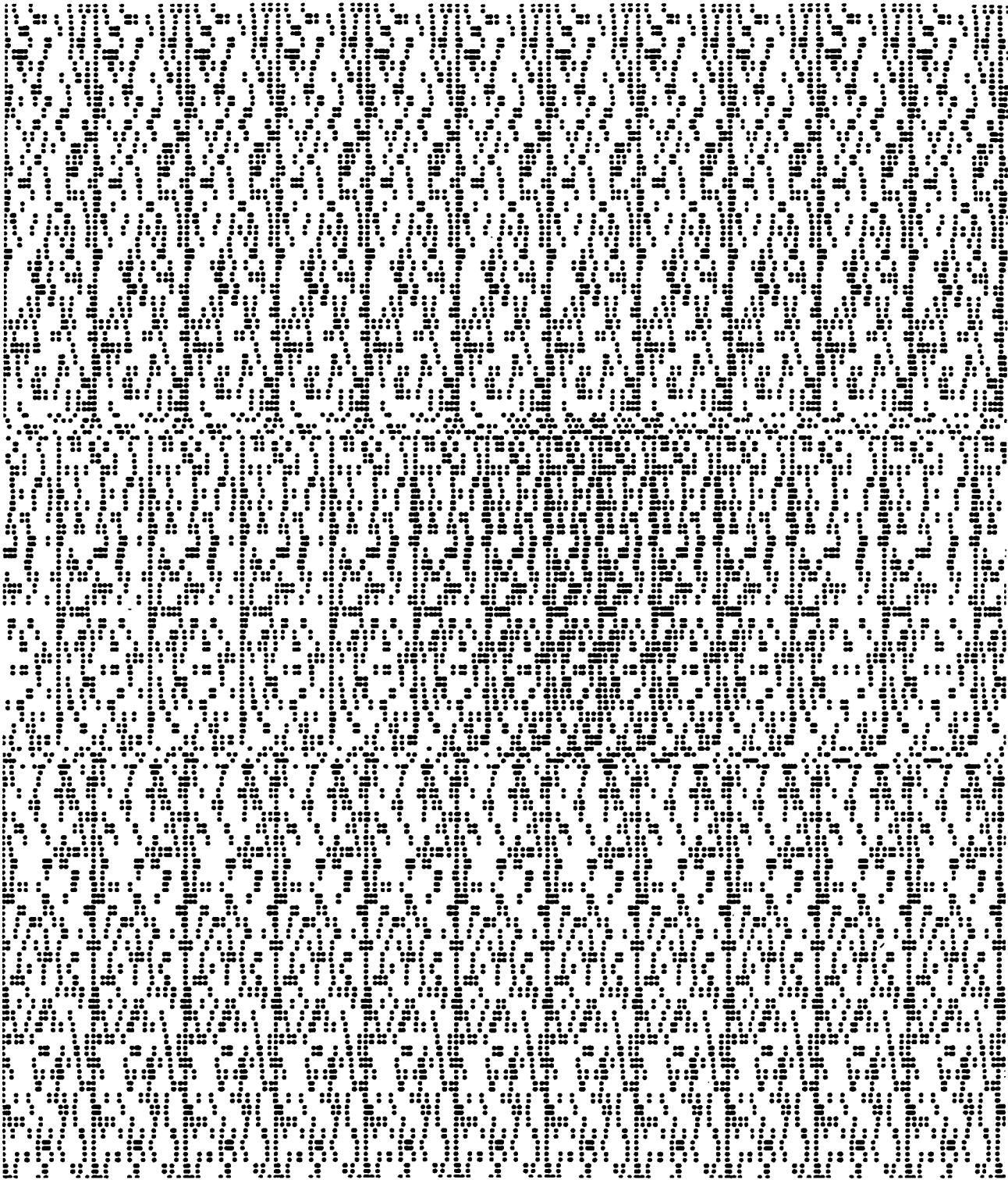
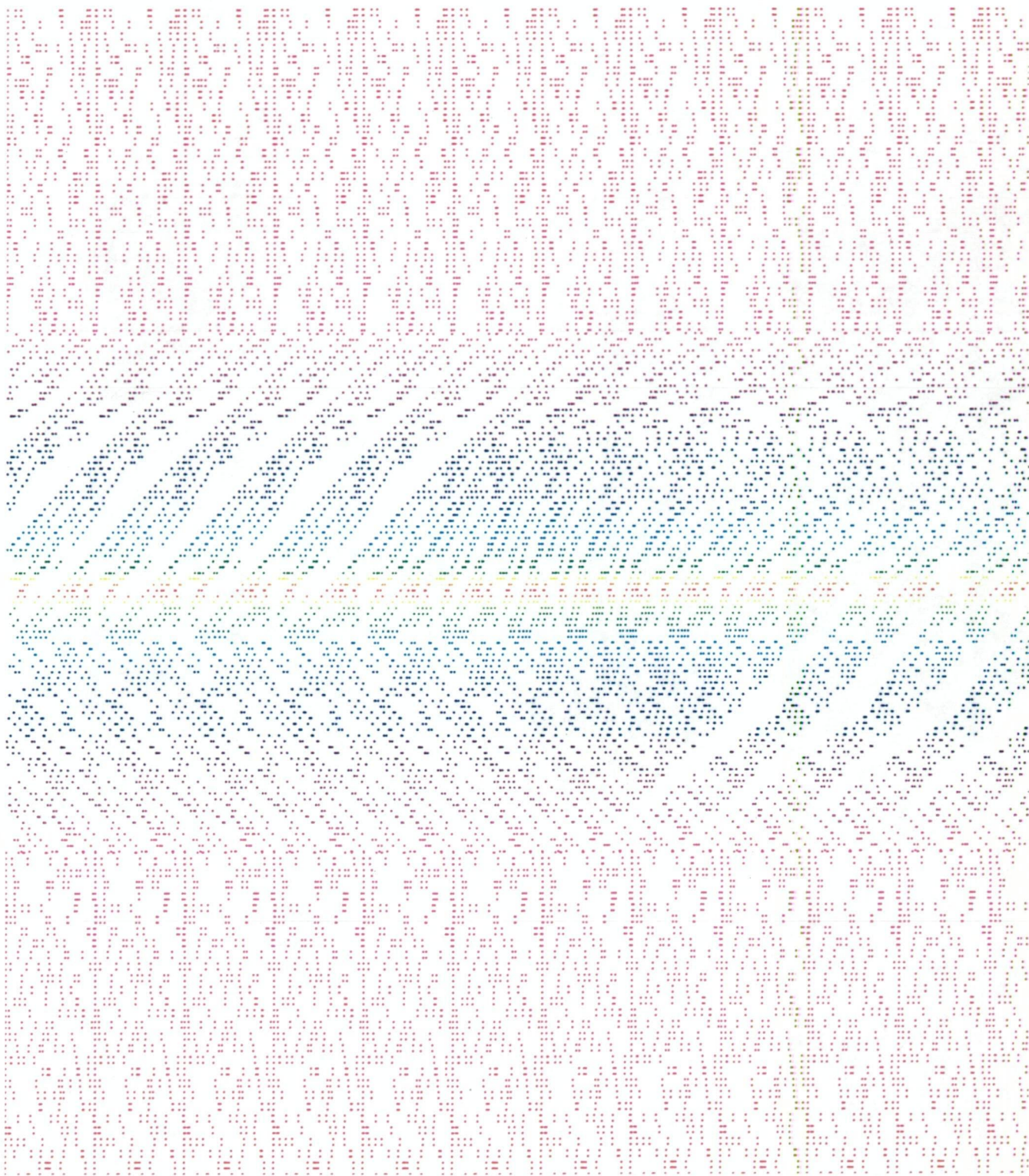


Figure 5.35: Two visible faces of a cube; viewpoint:(0, -30, 0); origin is at the centre of the page



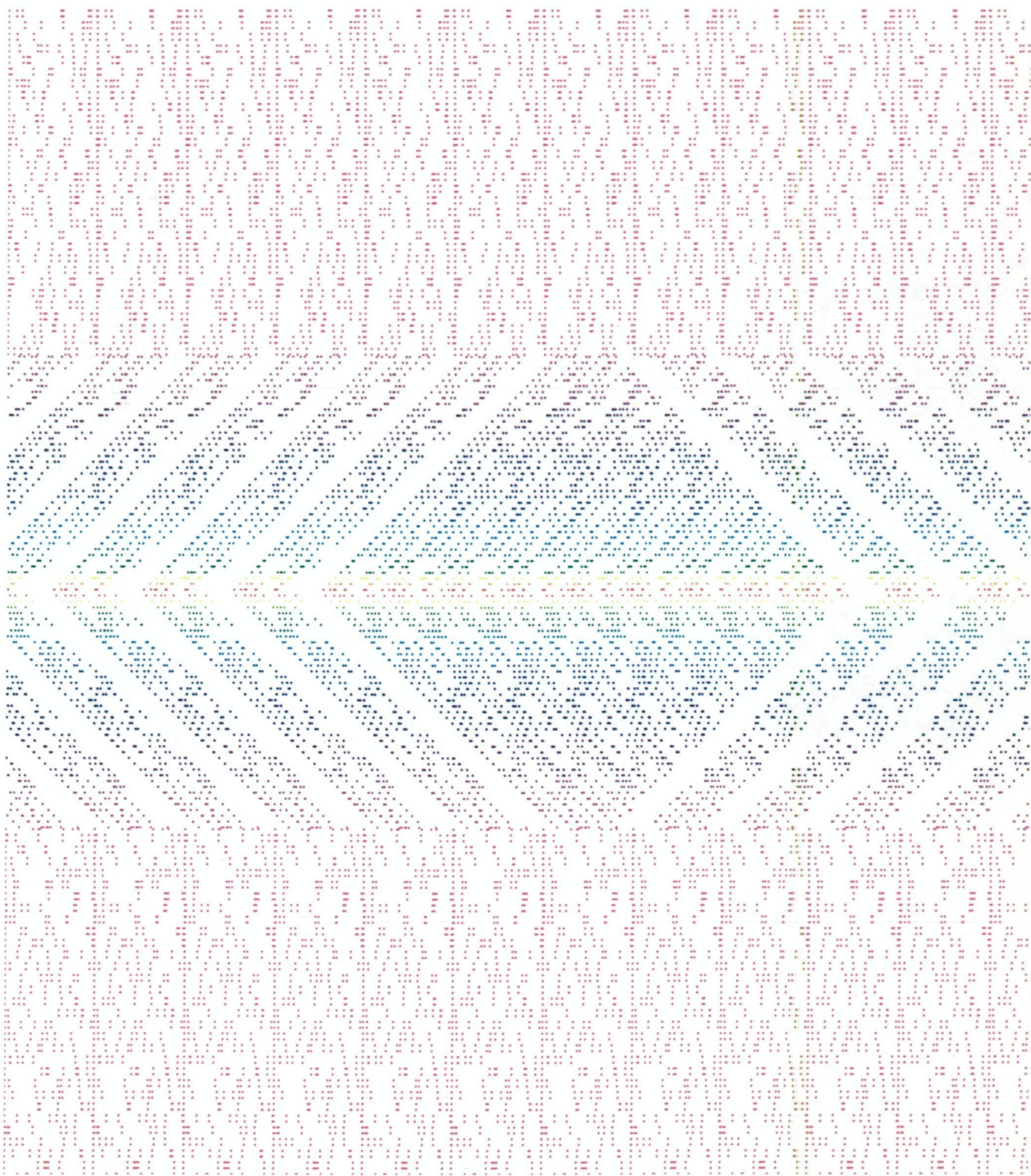


Figure 5.36: One visible face of a cube; viewpoint:(0, -30, 0); origin is at the centre of the page

5.4 Mixed Anamorphograms

We are now in a position to be able to combine all of our techniques to obtain anamorphograms which require a combination of the special viewing techniques. To illustrate various possibilities we will again examine a cube.

5.4.1 Combining two perspective drawings, and a Single-Image Stereogram, of the same cube

The perspective drawings of a cube from two viewpoints (one for each eye) are superimposed on the stereogram for this same cube in Figure 5.37.

We note that the correct focusing for the stereogram means that our eyes fuse the individual images of the two perspective drawings to give a three-dimensional cube outline which matches the perceived image of the stereogram.

5.4.2 Combining anaglyph and stereogram

The same example as that above can be used to combine the anaglyph technique with the stereogram as in Figure 5.38. The only difference, from the preceding case, being the two different colours used for the two perspective representations of the cube. We can view our combined picture with the appropriate spectacles with colour filters. The image obtained here is more satisfactory than that of Figure 5.37 since we are not aware of the individual pictures for each eye.

5.4.3 Combining a stereogram and a cylindrical mirror anamorphogram

Another possibility is the combining of a stereogram and a mirror distortion. We will aim to see a correct image of our stereogram in the mirror. Care must be taken here if we are to test our results as our image will need to be of suitable size for viewing and for apparently, 'fitting inside' our mirror. For convenience, its distorted picture must fit on our A4 page.

The following example of Figure 5.39 was created for a cylindrical mirror of

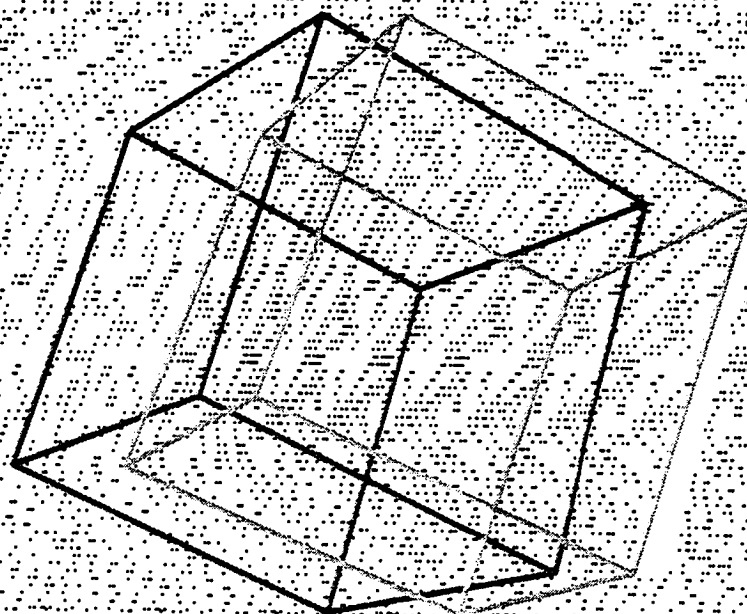


Figure 5.37: Three visible faces of a cube; viewpoints: $(-2.875, -30, 0)$ and $(2.875, -30, 0)$

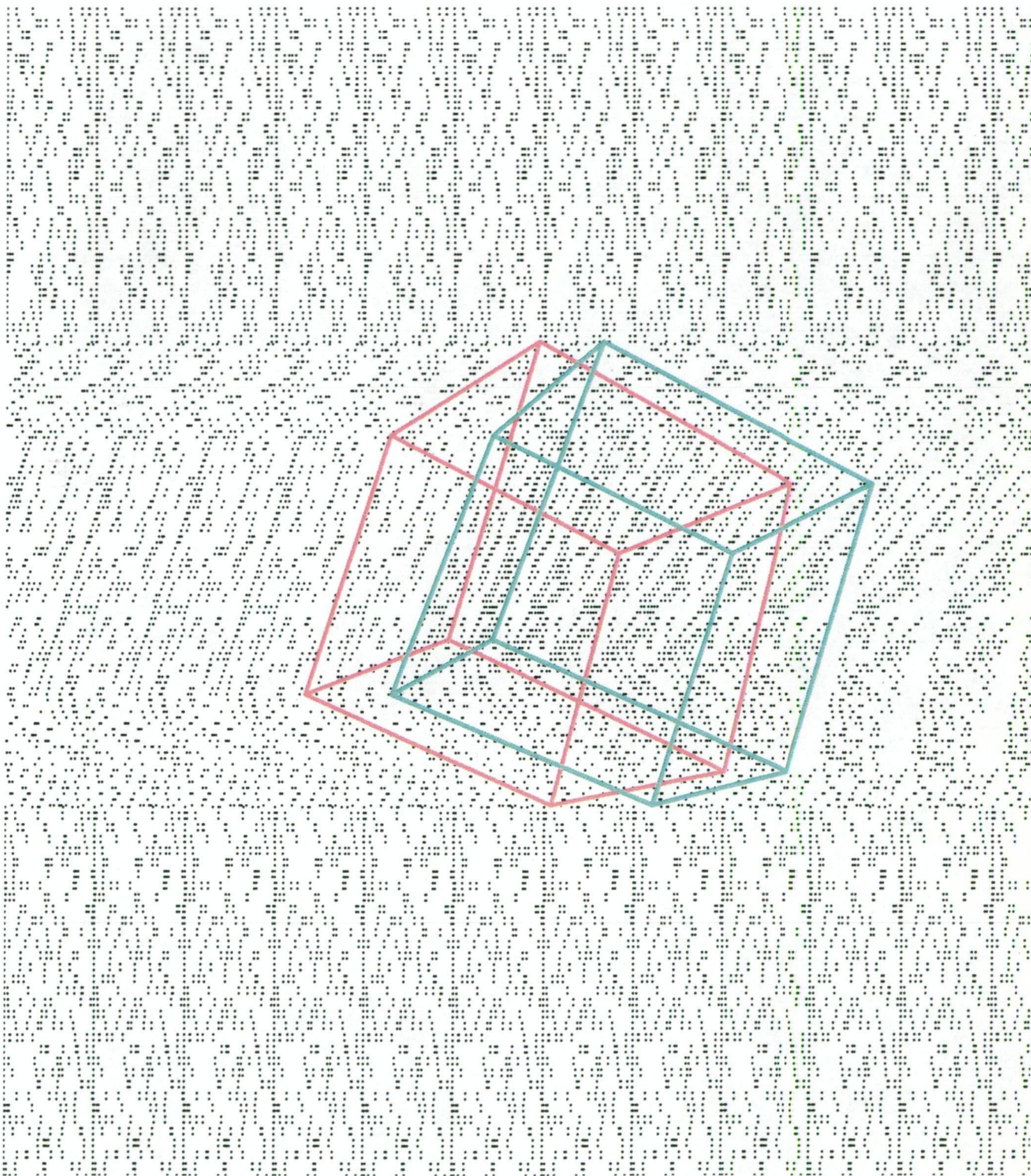


Figure 5.38: Three visible faces of a cube; viewpoints: $(-2.875, -30, 0)$ and $(2.875, -30, 0)$

radius 5cm. The viewpoint for the construction of the anamorphogram for the cylindrical mirror was $(0, -30, 30)$. If we are to view the 'image' in the mirror stereoscopically, then our viewpoints will not be precise. However for the practised stereogram viewer, and a good cylindrical mirror, the three-dimensional cube is clearly perceivable.

Another consideration is the choice of graphics elements used to represent our points on the stereogram. Even dots will be distorted by our mirror however, if we choose them to be small enough, then this distortion will be minimal. This is important for the matching of the correct pairs of dots by our eyes, although according to Julesz [13], it is possible for them to cope with very small variations in corresponding dot sizes.

We could also consider the case of a stereogram where the Dots are figures which can be represented in terms of co-ordinates. This would mean that application of the mirror program of Appendix D, which transforms co-ordinates to co-ordinates, will cause the Dots to be correctly distorted for our mirror. We have yet to try an example.

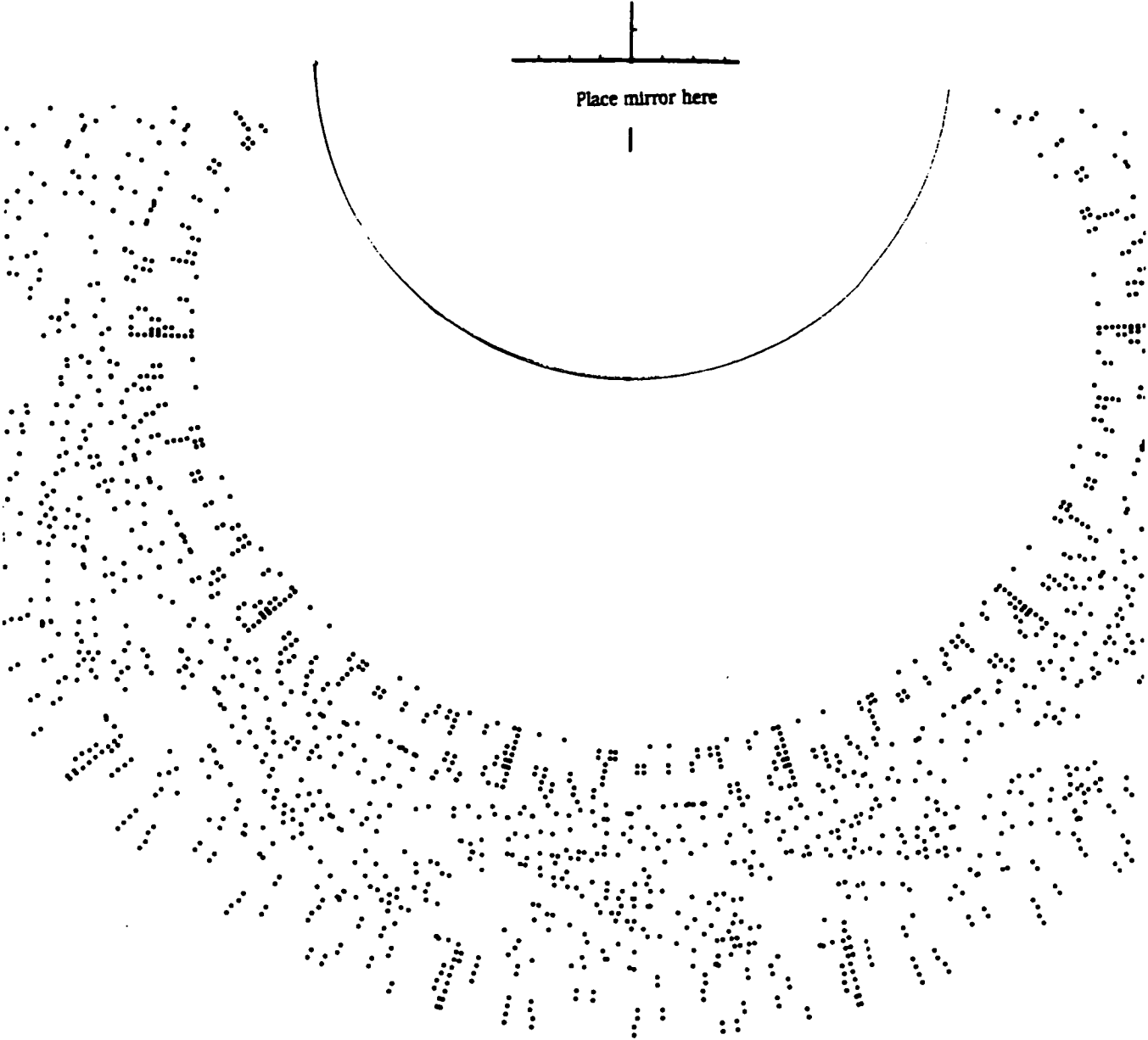


Figure 5.39: Three visible faces of a cube: viewpoints $(-2.875, -30, 0)$ and $(2.875, -30, 0)$, mirrorsurface: $x^2 + y^2 = 25$

Chapter 6

Single-Image Stereogram of a Special Sphere

In this chapter we consider an ‘alternative view’ of what we have discussed already. We begin with a brief description of some of the essential ideas of Projective geometry.

In Projective geometry we analyze the properties of geometric figures which remain invariant under projections. In order to do this we extend our Euclidean plane. We introduce the notions of ideal points and the ideal line. Each family of parallel lines in the plane defines a point called an *ideal point* (or *direction*). By family of parallel lines, we mean the collection of all lines parallel to a given line, L . We say that L and each line of the family meet at their associated ideal point. If we use the terminology of Row [26], and denote the ideal point associated with the line l as \hat{l} , then the statement “any two lines in the plane meet at one and only one point” can be represented as shown in Figure 6.1. The line which consists of all the ideal points

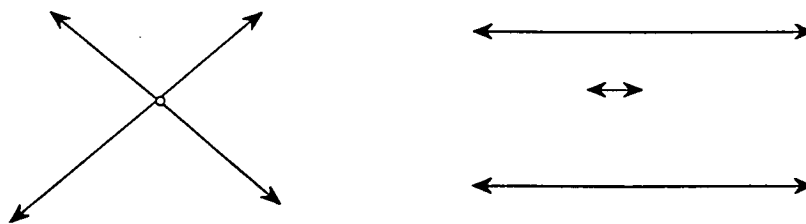


Figure 6.1: Any two non-parallel lines meet in a point and any two parallel lines meet at an ideal point

is called the *ideal line* or *line at infinity*.

We note that with the extension of the Euclidean plane to the *extended Euclidean* or *projective plane*, with the ideal line and ideal points, we have the *Principle of Duality*. This means that if each occurrence of the words 'point' and 'line' are interchanged in a geometrical statement, then the resulting statement is still true.

For example, we can say that:

Two *points* determine one and only one *line*, and

Two *lines* determine one and only one *point*.

Note: In Euclidean geometry two parallel lines do not determine a point.

Two such symmetrical statements and their corresponding geometric configurations are said to be *dual* results.

There are also other expressions that need to be interchanged in order to obtain the dual statement. For example, 'lie on' with 'pass through'. This depends on the terminology used by a particular writer. For example, Veblen [29] simplifies the reciprocation by referring to, a point 'on' a line, and a line 'on' a point.

We will now review some basic results of Projective geometry which lead to some new results. These will allow us to construct the single-image stereogram of a *fixed sphere*. This sphere is fixed in the sense that, if it is viewed from the same viewpoints (one for each eye) with both the *crossed* and *uncrossed* viewing techniques of section 3.2.1, then the perceived figure in each case is a different view of the same sphere. For convenience, we will refer to the employment of both viewing techniques as *dual* viewing.

This is in contrast to the usual situation where the two viewing techniques allow the viewer to perceive similar, but different, images. For example, if we see the outside surface of part of a cube using the 'uncrossed' technique, then we usually see the inside surface of what appears to be a smaller, slightly distorted cube, when we switch to the 'crossed' technique. The reader may check the accuracy of this statement, by viewing the stereogram of Figure 5.35 using both techniques. Another way of expressing this, is to observe that if certain dots present collinear images for uncrossed viewing, then their images for crossed viewing are also collinear. Such an observation leads to the result of Theorem 6.17. The case for part of a cube, is represented diagrammatically in Figure 6.2 (a), where we are looking down on one viewing plane from above. Similarly, we have represented the case for part of a sphere in Figure 6.2 (b). Here, according to our diagram, the perceived

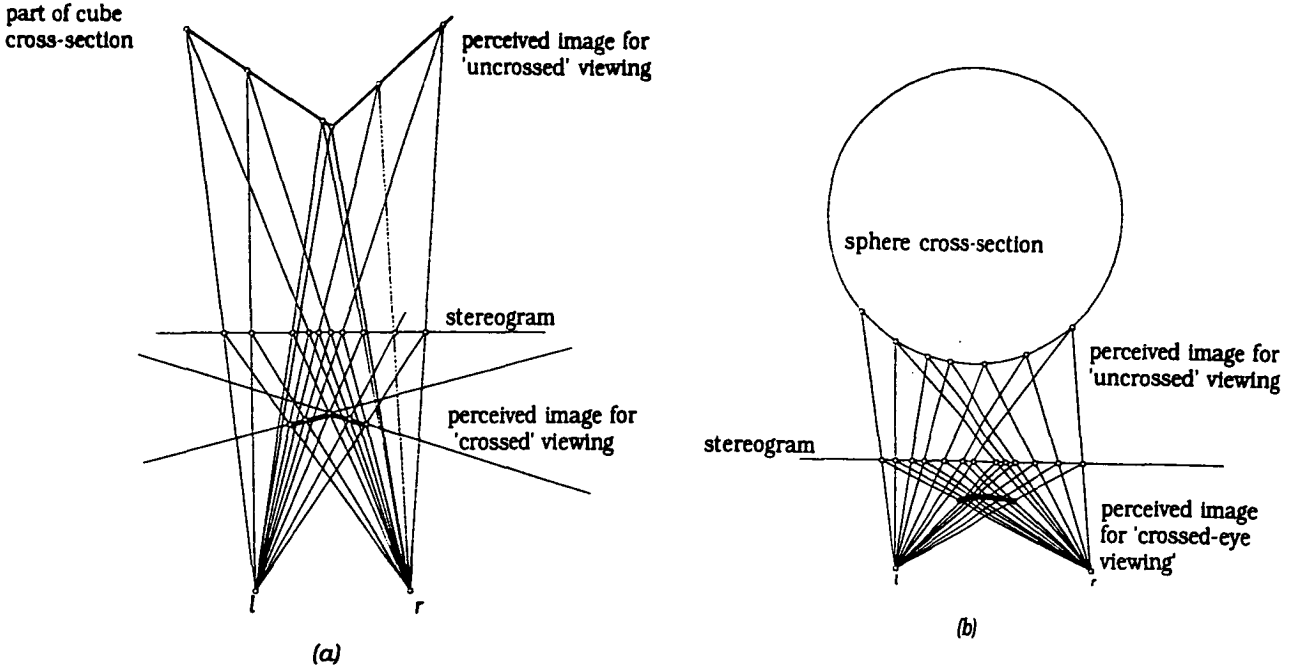


Figure 6.2: Viewing the same dots using both the crossed and uncrossed techniques

image for 'crossed-eye' viewing could be the inside of a flattened sphere such as an ellipsoid?

6.1 The Stereoscope and some related theorems

We begin by defining a *stereoscope* which consists of a given line D , co-planar with points l and r , where $l, r \notin D$, and a permutation, $\mu : D \rightarrow D$, of the points of D . We denote this stereoscope by $S(D, l, r, \mu)$. Associated with it we have a collection of points

$$\{x = (a \vee l) \wedge (\mu a \vee r) : a \in D, a \notin l \vee r\}.$$

Such a point, x , is illustrated in Figure 6.3 and will be referred to as the *image* of a and μa . It must be noted that $a \notin l \vee r$ ensures that the stereoscope is well-defined. $S(D, l, r, \mu)$ has an associated *inverse* which we write as

$$S^{-1}(D, l, r, \mu) = S(D, l, r, \mu^{-1})$$

where the associated collection of points is

$$\{x^{-1} = (b \vee l) \wedge (\mu^{-1} b \vee r) : b \in D, b \notin l \vee r\},$$

$$\text{or } \{x^{-1} = (\mu a \vee l) \wedge (a \vee r) : a \in D, a \notin l \vee r\}.$$

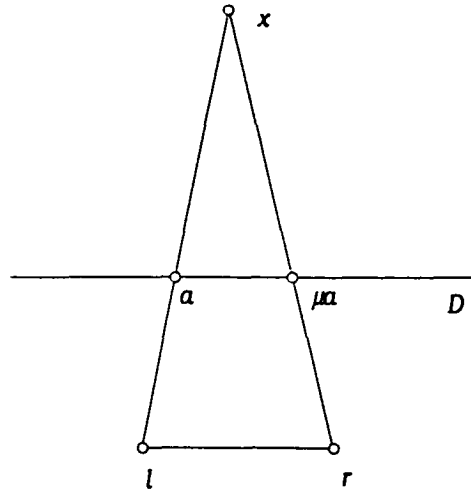


Figure 6.3: A stereoscope presenting the image, x , of points a and μa .

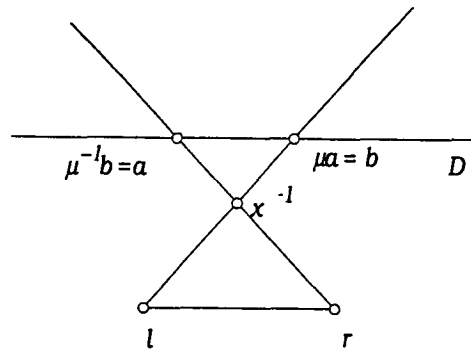


Figure 6.4: Inverse stereoscope

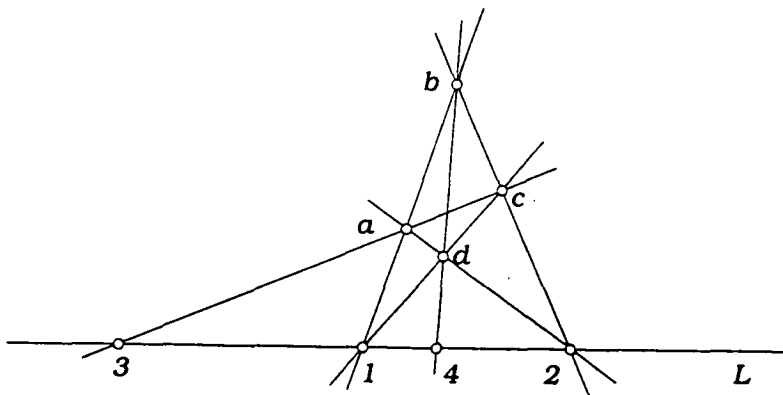


Figure 6.5: 4 is the harmonic conjugate of 3 with respect to 1 and 2

Such a point, x^{-1} , is illustrated in Figure 6.4. It will be referred to as the *inverse image* of points a and μa .

Important to our discussion is the notion of an *harmonic conjugate*. In Figure 6.5 we have four coplanar points a, b, c and d (no three of which are collinear) coplanar with the line L . We see that $a \vee b$ meets $c \vee d$ on L and $a \vee d$ meets $b \vee c$ on L . If we let $(a \vee b) \wedge (c \vee d) = 1$, $(a \vee d) \wedge (b \vee c) = 2$, $(b \vee d) \wedge L = 4$ and $(a \vee c) \wedge L = 3$, then the point 4 is called the *harmonic conjugate* of 3 with respect to 1 and 2 and is denoted by $4 = H(3; 12)$. We say that the four points, 1, 2, 3 and 4 are *harmonically related*. In fact we have the following theorems :

Theorem 6.1 *For any three collinear points 1, 2 and 3, there is a unique harmonic conjugate of 3 with respect to 1 and 2.*

Proof: see Pedoe [22, page 54]

◇

Theorem 6.2 *If 1 and 2 are harmonic conjugates with respect to 3 and 4 then 3 and 4 are harmonic conjugates with respect to 1 and 2.*

Proof: see Veblen [29, page 81]

◇

Theorem 6.3 *Let 1 and 2 be any two distinct points and 3 the harmonic conjugate of the ideal point of the line $1 \vee 2$ with respect to 1 and 2. Then 3 is the midpoint of the segment $\overline{1 \vee 2}$.*

Proof: see Veblen [29, page 80] vol. 2

◇

In general, if we denote the n^{th} image, in our stereoscope $S(D, l, r, \mu)$, of a and μa as x^n , which corresponds to the image of points $\mu^{n-1}a$ and $\mu^n a$, then we have Figure 6.6. Here we see that d is the harmonic conjugate of $d' = D \wedge (l \vee r)$ with respect to l and r .

It must be noted that when $l \vee r$ is parallel to D , then d (by Theorem 6.3) is the midpoint, m , of l and r , which in this particular case means that m is the harmonic conjugate of the ideal point on $l \vee r$ with respect to l and r .

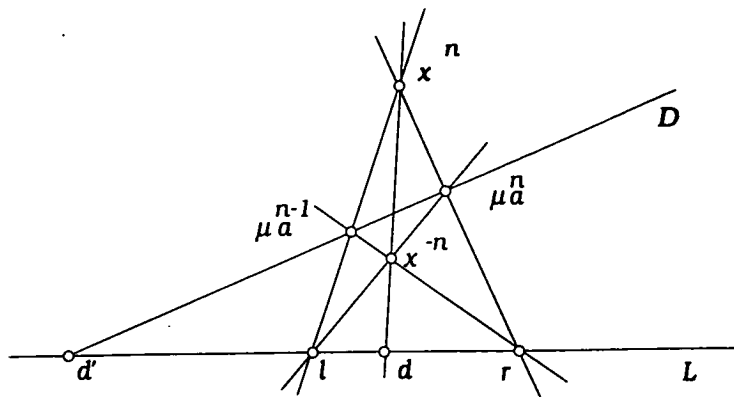


Figure 6.6: Stereoscope of the n^{th} image of a and μa

Theorem 6.4 *If x^n is the n^{th} image of points a and μa , of Figure 6.6, and x^{-n} is their n^{th} inverse image, then x^n and x^{-n} are collinear with d , $\forall a \in D$.*

Proof: This follows from the definition of harmonic conjugate and the fact that d depends only on d' , l and r . \diamond

Given a figure F and a point p , every point of F distinct from p determines with p a line, and every line of F not on p determines with p a plane. The set of these lines and planes through p is called the *projection* of F from p . The individual lines and planes of the projection are also called the *projectors* of the respective points and lines of F .

Two figures F_1 and F_2 are said to be in *1:1 correspondence* if every element of F_1 corresponds to a unique element of F_2 in such a way that every element of F_2 is the correspondent of a unique element of the figure F_1 . A figure is said to be in *1:1 correspondence with itself*, if every element of the figure corresponds to a unique element of the same figure in such a way that every element of the figure is the correspondent of a unique element. Two elements that are associated in this way are said to be corresponding or *homologous* elements. An example is shown in Figure 6.7.

If we now consider any line, L , and any point, $l \notin L$ (as shown in Figure 6.8), then an *elementary map* θ , is a particular 1:1 correspondence between all the lines through l and all the points, p_i , on L .

That is, using the notation of Row [26], $\theta : p_i \leftrightarrow l \vee p_i$ and we denote this

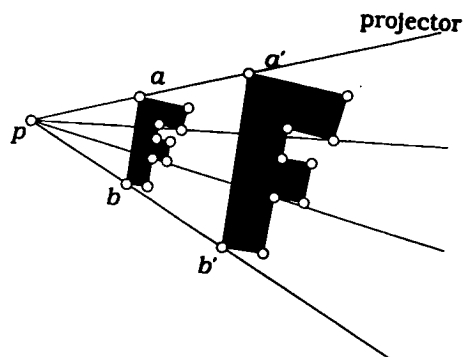


Figure 6.7: Homologous elements in two corresponding figures

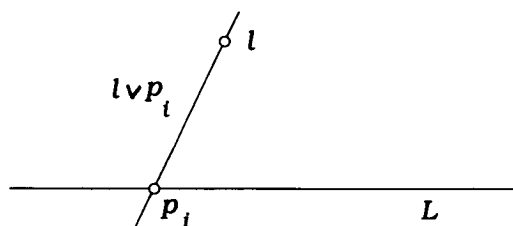


Figure 6.8: Elementary maps

map by

$$L \overset{\text{el}}{\bar{\wedge}} l \text{ with inverse map } l \overset{\text{el}}{\bar{\wedge}} L.$$

A *pencil of lines* is the figure formed by the set of all co-planar lines through the same point. This point is called the *vertex* or *centre* of the pencil. If the vertex is l , and the set of lines $\{P_1, P_2, P_3, \dots\}$ is a subset of the pencil or *sub-pencil*, we denote the subset with the centre by $l(P_1, P_2, P_3, \dots)$.

The figure formed by the set of all points on the same line, say L , is called a *range* of points on L and a subset of the range on L is denoted by $L(p_1, p_2, p_3, \dots)$.

Two ranges in the same plane are *perspective* provided every two homologous points of the ranges are on a line of a pencil of lines through a point. This means that these points have the same projection from this point. For example, in Figure 6.9 the ranges on L and L' are perspective, and we denote the 1:1 correspondence between the ranges by

$$L(p_1, p_2, p_3, \dots) \overset{l}{\bar{\wedge}} L'(p'_1, p'_2, p'_3, \dots).$$

Such a mapping is a composition of two elementary maps

$$L(p_1, p_2, p_3, \dots) \overset{\text{el}}{\bar{\wedge}} l \overset{\text{el}}{\bar{\wedge}} L'(p'_1, p'_2, p'_3, \dots)$$

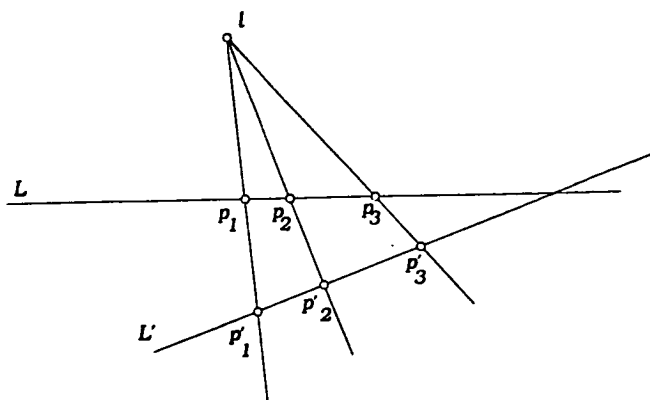


Figure 6.9: A perspectivity between points on L and L' , with centre l .

and establishes a (1:1) correspondence between the points on L and those on L' and is called a *perspectivity*. The point, l , is called the *centre of perspectivity*.

Similarly, we can have the dual situation; Two co-planar pencils of lines are *perspective*, if every two homologous lines intersect in a point of the same range of points. That is, in Figure 6.10 the pencils through l and l' are said

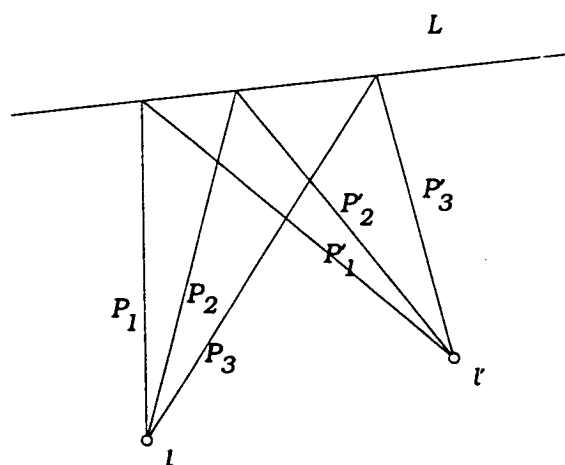


Figure 6.10: A perspectivity between lines through l and l' , with axis L .

to be perspective, and we write

$$l(P_1, P_2, P_3, \dots) \overset{L}{\bar{\wedge}} l'(P'_1, P'_2, P'_3, \dots).$$

Such a mapping is a composition of two elementary maps

$$l(P_1, P_2, P_3, \dots) \overset{\text{el}}{\bar{\wedge}} L \overset{\text{el}}{\bar{\wedge}} l'(P'_1, P'_2, P'_3, \dots)$$

and establishes a (1:1) correspondence between the lines through l and those through l' and is called a *perspectivity*. The point, L , is called the *axis of perspectivity*.

We define a *projectivity* between the points of two lines L and L'' , or between the lines through two points l and l'' , to be a composition of a finite sequence of perspectivities. As an example, consider the situation shown in Figure 6.11, where the line L'' must cut L' but need not lie in the plane $l \vee L$. The

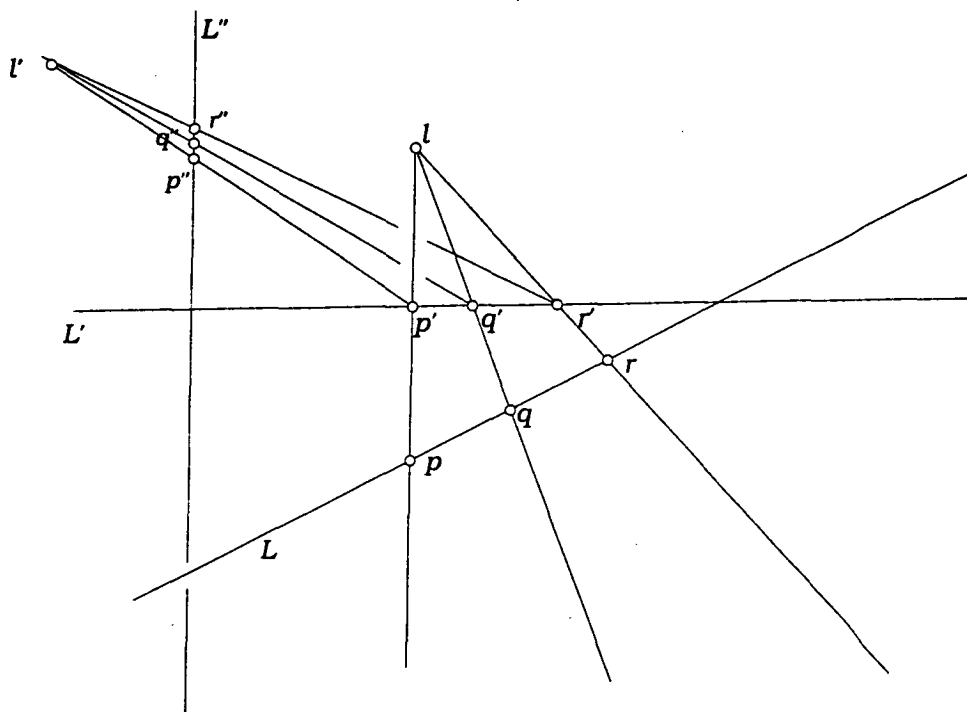


Figure 6.11: Projectivity between ranges on L and L''

sequence of perspectivities is

$$L(p, q, r, \dots) \stackrel{l}{\bar{\wedge}} L'(p', q', r', \dots) \stackrel{l''}{\bar{\wedge}} L''(p'', q'', r'', \dots).$$

We now have a (1:1) correspondence between the ranges of L and L'' which is described as a projectivity and is written

$$L(p, q, r, \dots) \bar{\wedge} L''(p'', q'', r'', \dots).$$

Similarly, we can have the dual which is a projectivity between the pencils of l and l'' as illustrated in Figure 6.12. In this example the sequence of projectivities is

$$l(P, Q, R, \dots) \stackrel{L}{\bar{\wedge}} l'(P', Q', R', \dots) \stackrel{L''}{\bar{\wedge}} l''(P'', Q'', R'', \dots).$$

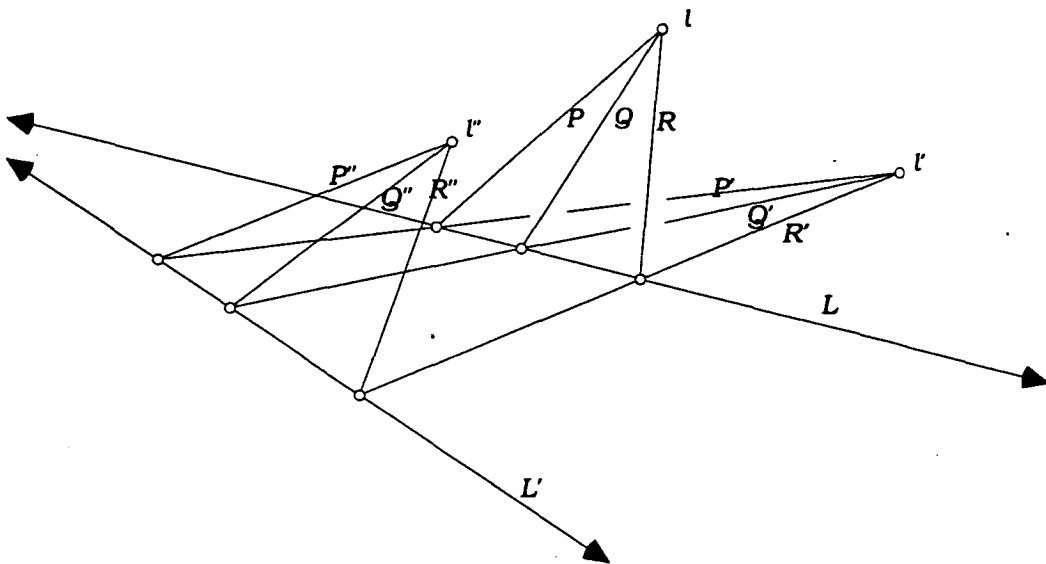


Figure 6.12: Projectivity between pencils through l and l''

and the projectivity is written

$$l(P, Q, R, \dots) \bar{\wedge} l''(P'', Q'', R'', \dots).$$

We will represent projectivities by Greek letters. If a projectivity, η , maps every line of a pencil with vertex l onto a line of a pencil with vertex l'' , then we will denote such a mapping by

$$\eta : l \bar{\wedge} l'' \text{ or } l \bar{\overset{\eta}{\wedge}} l''.$$

If η maps a line L onto a line L'' , then we may denote L'' by ηL . We will denote ' L is a line through the point l ' by $L \ni l$.

6.2 Introducing Conics

Central to our discussion is the concept of a *conic*. Consider two pencils of lines in the same plane and with vertices l and r and suppose that these pencils are in *projective correspondence*. That is, \exists a projectivity $\eta : l \bar{\wedge} r$. As illustrated in Figure 6.13, to any line $l \vee p$ of the first pencil there is by the correspondence a unique line $r \vee p$ of the second pencil; and conversely. The locus of the points of intersection, p , of the corresponding lines of the

two pencils is called a *conic*. The vertices of the pencils, l and r , are called the *generating bases* of the conic. It must be noted that we have considered

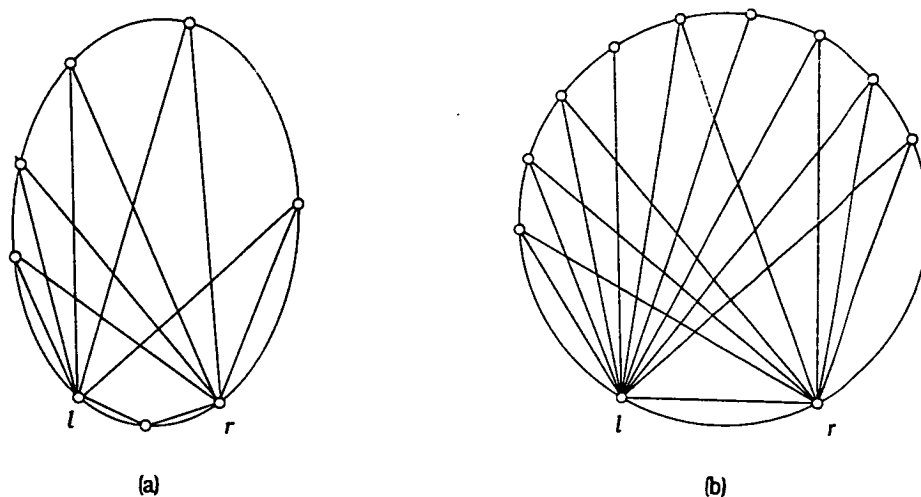


Figure 6.13: Conic with generating bases l and r

an example of such a projective correspondence already. This was for the Vieth-Muller circle obtained in Luneburg's experiment of subsection 3.5.2. That is, if the angles between neighbouring lines of the corresponding sub-pencils are equal, the angle being the same for each sub-pencil, the conic is a circle, as illustrated in Figure 6.13 (b).

A conic may be either *singular* or *nonsingular*. To understand what these terms mean, we suppose $\theta : a \bar{\wedge} b$ is a projectivity defining the conic given by $\{x/x \in X \ni a \text{ and } x \in \theta X\}$. If θ is a perspectivity, $\theta : a \bar{\wedge}^L b$, then the conic is the set of points of L together with the set of points of $a \vee b$. In this case the conic is described as being *singular*. If θ is not a perspectivity and $a \neq b$ then the conic is *non-singular*.

We now state two important theorems of Projective geometry; Desargue's Theorem and Pappus' Theorem. The proofs may be found in Veblen [29]. Their results are applied in the proof of the next lemma.

Theorem 6.5 Desargues:

If two co-planar triangles are perspective from a point, the three pairs of homologous sides meet in collinear points. (see Figure 6.14)

◇

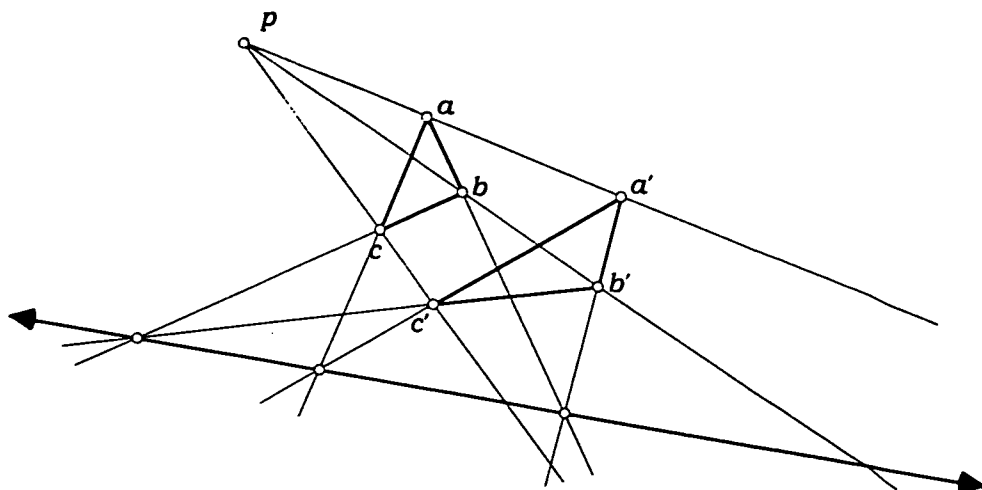


Figure 6.14: Triangles abc and $a'b'c'$ are perspective from p

Theorem 6.6 Pappus:

If alternate vertices of a planar hexagon lie on two lines, the three pairs of opposite sides meet in three collinear points. (see Figure 6.15)

◇

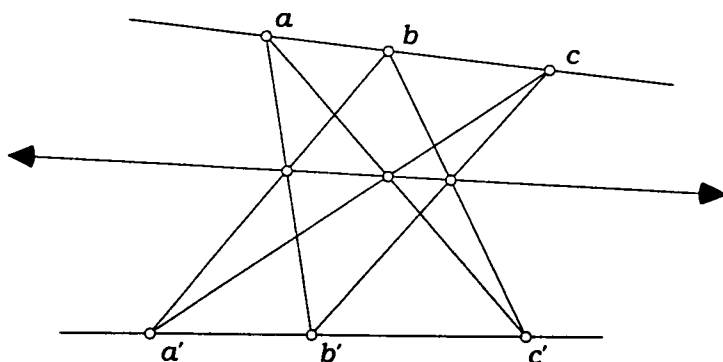


Figure 6.15: Hexagon abc and $a'b'c'$; opposite sides are $a \vee b'$, $a' \vee b$, $b \vee c'$, $b' \vee c$, $a \vee c'$, $a' \vee c$

Lemma 6.7 Suppose $\theta : a \bar{\wedge} b$ is a projectivity between the lines through a point a and a point b distinct from a , then θ is a perspectivity if and only if $\theta(a \vee b) = a \vee b$.

Proof: Suppose θ is a perspectivity with axis L . That is, $\theta : a \bar{\wedge}^L b$ and $\forall X \ni a, \theta X = (X \wedge L) \vee b$. Therefore if $X = a \vee b$ then $\theta X = [(a \vee b) \wedge L] \vee b = a \vee b$.

Conversely, suppose $\theta(a \vee b) = a \vee b$. Any line $X \ni a$ can be mapped onto a line $Y \ni b$ by one or a composition of two perspectivities (Theorem 6.9). Suppose that there are two, and denote the intermediate vertex by c , and the axes by R and S . That is, $\theta : a \stackrel{R}{\wedge} c \stackrel{S}{\wedge} b$. Now there are 2 cases to consider:

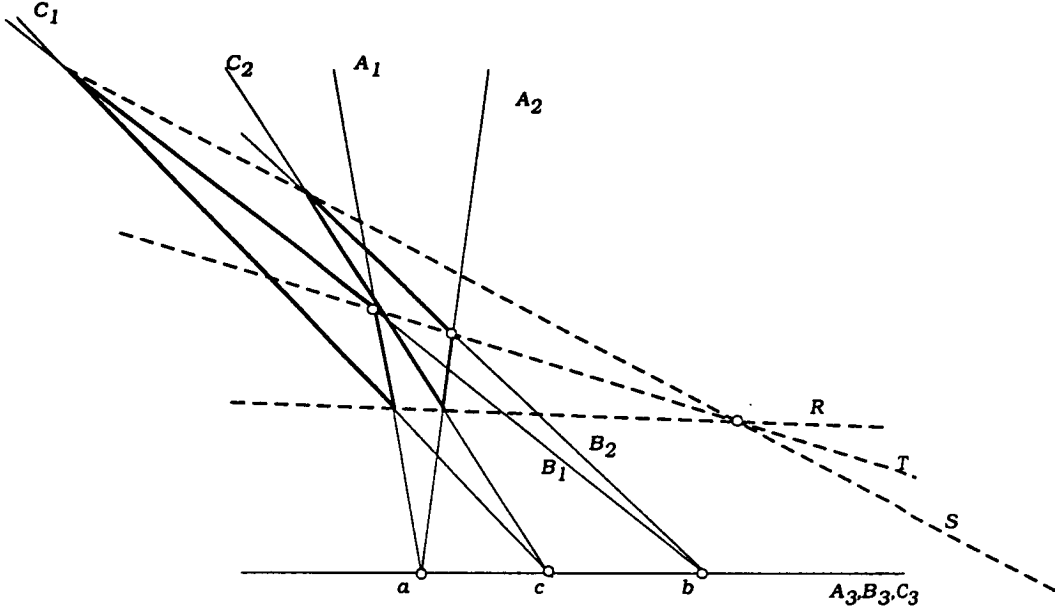


Figure 6.16: The points a, c and b are collinear

(i) a, c, b are collinear $\Rightarrow \exists$ a line T such that R, S, T are concurrent and $a \stackrel{T}{\wedge} b$. This follows from

$$a(A_1, A_2, A_3, \dots) \stackrel{R}{\wedge} c(C_1, C_2, C_3, \dots) \stackrel{S}{\wedge} b(B_1, B_2, B_3, \dots)$$

where we consider triangles $A_1B_1C_1$ and $A_2B_2C_2$ of Figure 6.16. Now $A_1 \wedge A_2, C_1 \wedge C_2$ and $B_1 \wedge B_2$ are collinear points and hence by the dual of Desargue's Theorem 6.5, the join of $A_1 \wedge B_1$ and $A_2 \wedge B_2$ (which we have called T) is concurrent with R and S .

(ii) a, c, b are not collinear: Suppose that $R \neq S$ and let $P = a \vee c$ and $Q = c \vee b$, $P \neq Q$ as shown in Figure 6.17.

Let $a \vee b = F$. This is the given fixed line of our projectivity. Therefore F, R and S are concurrent.

$$Q' = a \vee (R \wedge Q)$$

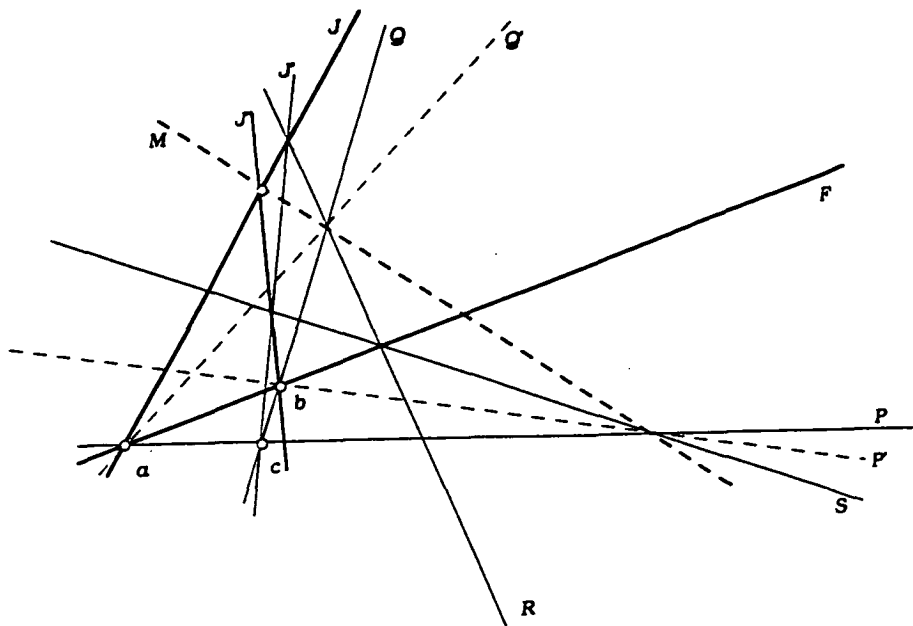


Figure 6.17: The points a, c and b are not collinear

$$P' = b \vee (S \wedge P)$$

$$M = (S \wedge P) \vee (R \wedge Q)$$

Consider any line J through a , $J \ni a$, and its images J' and J'' under the two perspectivities. The hexagon $RJ'SPFQ$ and the dual of Pappus' Theorem 6.6 give J, J'', M are concurrent, where

$$J'' = (F \wedge Q) \vee (S \wedge J')$$

$$J = (P \wedge F) \vee (R \wedge J')$$

Therefore $a \bar{\wedge} b$ is the perspectivity $a \stackrel{M}{\bar{\wedge}} b$. ◇

Next we need some well-known theorems on projective correspondences.

Theorem 6.8 *Given two triplets of distinct points, p_1, q_1, r_1 and p_3, q_3, r_3 lying, respectively, on two distinct lines, L_1 and L_3 , there is a projectivity which assigns p_1, q_1, r_1 to p_3, q_3, r_3 respectively.*

Proof: see Pedoe [22, page 45]. ◇

In general we have the following Theorem and its Corollary. The proofs may be found in Pedoe [22, pages 46-52].

Theorem 6.9 *If a pencil of lines with vertex, a , is related by a chain of, say m , perspectivities to a pencil of lines with vertex, $b \neq a$, then this projectivity is equivalent to at most two perspectivities.*

◇

Corollary 6.10 *A 1:1 projective correspondence between two ranges on the same straight line can be obtained as the result of at most three perspectivities.*

◇

The fact that a projectivity between the points of two lines (which may coincide) is uniquely determined by the assignment of three pairs of corresponding points comes from the following result. The proof may be found in Pedoe [22, page 64].

Theorem 6.11 *A necessary and sufficient condition that a projectivity between two ranges is uniquely determined by the assignment of three pairs of corresponding points is that Pappus' theorem holds.*

◇

Consider the stereoscope, $S(D, l, r, \mu)$, such as the one represented by Figure 6.18, where $p' = \mu p, q' = \mu q, s' = \mu s$; a, b, c are the perceived images of p and p', q and q', s and s' respectively.

We have the following results:

Theorem 6.12 *If the images a, b, c are collinear then the permutation $\mu : D \rightarrow D$ is a projectivity.*

Proof: a, b, c are collinear $\Rightarrow \mu$ can be written

$$(p, q, s) \stackrel{l}{\wedge} (a, b, c) \stackrel{r}{\wedge} (p', q', s')$$

where $(p, q, s), (a, b, c)$ and (p', q', s') are ranges on $D, a \vee b$ and D respectively. Hence μ is a projectivity by definition. ◇

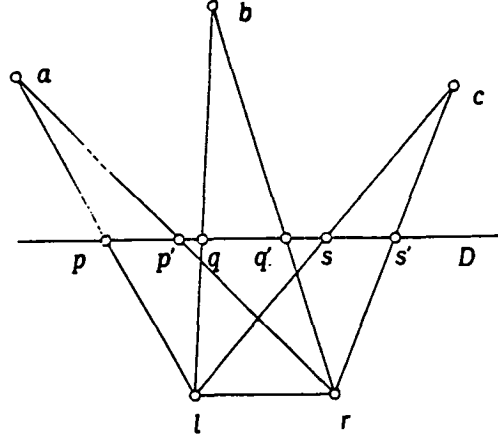


Figure 6.18: The points p, p', q, q', s, s' represent the positions of corresponding pairs of dots of a stereogram

Lemma 6.13 *The permutation, $\mu : D \rightarrow D$ is a projectivity if and only if the stereoscope is a conic containing its generating bases l and r .*

Proof: μ is a projectivity $\Rightarrow l \overset{\text{el}}{\bar{\wedge}} D \overset{\mu}{\bar{\wedge}} D \overset{\text{el}}{\bar{\wedge}} r$ is a projectivity since it is the composition of a finite number of perspectivities; call it β . It defines a conic with generating bases l and r which is the stereoscope. Any point, x , on this conic is defined by

$$x = (l \vee a) \wedge (r \vee \mu a)$$

for some $a \in D$.

Conversely, suppose we have a conic defined by $\beta : l \overset{\text{el}}{\bar{\wedge}} D \overset{\text{el}}{\bar{\wedge}} r$ which we can also write as $\beta : l \overset{\text{el}}{\bar{\wedge}} D \overset{\text{el}}{\bar{\wedge}} D \overset{\text{el}}{\bar{\wedge}} r$, then any point on this conic is defined by $(l \vee d) \wedge (r \vee \phi d)$, where ϕd is a point on D determined uniquely by the conic point. This means we have a projectivity $\phi : D \bar{\wedge} D$ and since this projectivity maps $a \rightarrow \mu a$ for any $a \in D$, $\phi = \mu$ (the defining projectivity for our stereoscope). \diamond

Lemma 6.14 *The stereoscope conic described in Lemma 6.13 is :*

(i) *singular if and only if $\mu[(l \vee r) \wedge D] = (l \vee r) \wedge D$*

That is, in Figure 6.19, $\mu(d') = d'$ and in this case the stereoscope is a line S . The fixed points of μ are $S \wedge D$ and $(l \vee r) \wedge D$. There is exactly one fixed point if and only if S, D and $l \vee r$ are concurrent.

(ii) *non-singular if and only if $\mu[(l \vee r) \wedge D] \neq (l \vee r) \wedge D$. That is, $\mu(d') \neq d'$*

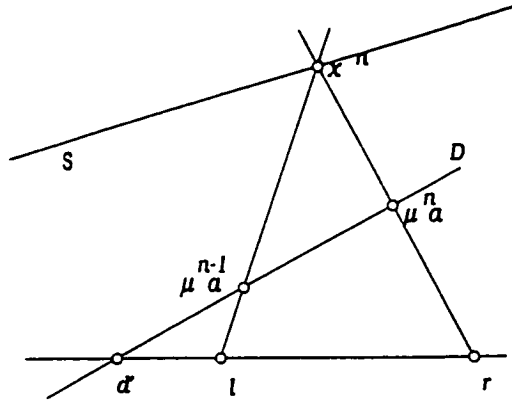


Figure 6.19: Singular conic: d' fixed by μ ; S is a line

Proof: (i) The stereoscope conic is singular $\Leftrightarrow \beta : l \bar{\wedge} r$ is a perspectivity (by definition) - with axis S , say, and then the conic is the set of points on the line S together with the points of $l \vee r \Leftrightarrow \beta(l \vee r) = (l \vee r)$ (by Lemma 6.7).

Any point, x , on this conic is defined by

$$x = (l \vee a) \wedge (r \vee \mu a)$$

for some $a \in D$. Therefore, if $x \in D$ then $a = \mu a$. Now $(l \vee r) \wedge D$ and $S \wedge D$ are points of D and so are fixed points of μ . That is, we have two possible cases:

- (a) there are two fixed points of μ [see Figure 6.20(a)] and
- (b) there is one fixed point of μ [see Figure 6.20(b)] when $S, D, l \vee r$ are concurrent.

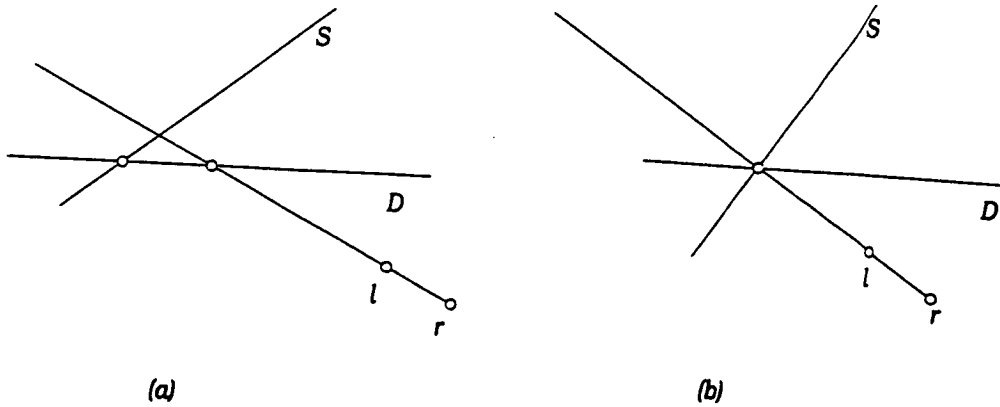


Figure 6.20: (a) two fixed points and (b) one fixed point

(ii) The stereoscope conic is nonsingular $\Leftrightarrow \beta : l \bar{\wedge} r$ is not a perspectivity $\Leftrightarrow \beta(l \vee r) \neq l \vee r$ (from Lemma 6.7). That is, $\beta(l \vee r) = r \vee p$ for some line $r \vee p$, distinct from $l \vee r$, through r . Now by definition of a conic, the point $(l \vee r) \wedge \beta(l \vee r) = (l \vee r) \wedge (r \vee p) = r$ lies on the conic. This point can also be represented as $(l \vee d) \wedge (r \vee \mu d)$ for some $d \in D$; in particular, the point $d' = (l \vee r) \wedge D$. Therefore, $\mu(d') \neq d'$ otherwise, $l \vee r$ and $r \vee p$ would meet on D which is impossible unless they are the same line. \diamond

In relation to Single-Image Stereograms, a consequence of Lemma 6.14 is that we will perceive a line S which is parallel to both the stereogram and the viewer, exactly when the only fixed point of the permutation μ is the ideal point containing the lines $l \vee r$, D and S . This is another way of representing our result of Theorem 3.1 which explains our perception of dots on a line which is parallel to a row of equally-spaced dots on a stereogram. We will see that the permutation μ is a translation of the points of D , in this special case. That is, for all $p, q \in D$, $|p \vee \mu p| = |q \vee \mu q|$. Recall from equation 3.1 that we have an expression representing the distance between a and μa , $a \in D$. Since the distance would generally be expected to vary with a , we will write it as $s(a) = p_d(a)e/(p_d(a) + d)$. In our particular case, all the perceived points are on the line S , parallel to D . Hence p_d is fixed, and μ is a translation. An example is shown in Figure 6.21, where $\mu a = b$ and $\mu b = c$. That is, b must be the midpoint of $\overline{a \vee c}$. By Theorem 6.3, b is

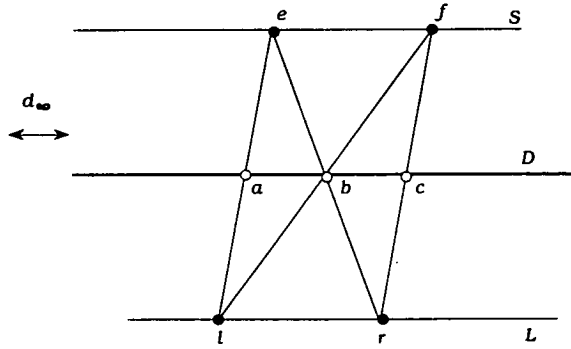


Figure 6.21: $efrl$ is a 4-pt circuit: pairs of opposite sides are $e \vee f$ and $l \vee r$, $e \vee r$ and $l \vee f$, $e \vee l$ and $f \vee r$

indeed the midpoint since if we consider the 4-pt circuit $efrl$, $b = H(d_\infty; ac)$, where d_∞ is the ideal point of the line D through a and c .

A *collineation* is a 1:1 mapping, $\Theta : \Pi \rightarrow \Pi$, of a plane Π onto itself, in which points are mapped onto points, lines are mapped onto lines, and incidence properties are preserved. That is, if L is a line on Π ($L \in \Pi$) and p is a point on L ($p \in L$) then $\Theta L \in \Pi$ and $\Theta p \in \Theta L$.

A *central collineation* in a plane is a projective collineation leaving fixed every point on a given line, L , and every line through a given point, p . The line, L , and the point, p , are called the *axis* and *centre*, respectively, of the central collineation which is denoted by $\Theta = (p, L)$ - collineation. If the centre does not lie on the axis then the collineation is called a *planar homology*; if $p \in L$ then it is a *planar elation*. We have the following result:

Theorem 6.15 *There are no other fixed elements besides L and the points on it, and p and the lines through it, unless Θ is the identity transformation.*

Proof: see Pedoe [22, page 102] ◇

A *central involution*, $\Theta : \Pi \rightarrow \Pi$, is a central collineation of order 2. That is, $\Theta(\Theta(a)) = \Theta^2(a) = a$, $\forall a \in \Pi$.

Theorem 6.16 *A central (p, L) -collineation, θ , in a plane is uniquely determined if the centre, axis and any two homologous points (not on the axis or centre) are given, with the restriction that the homologous points must be collinear with p .*

Proof: If there are two (p, L) -collineations α and β which map a point $v (\notin L, \neq p)$ onto αv , then the collineation $\beta^{-1}\alpha$ would have v as a fixed point. Since there can be no fixed point other than p and the points on L (by Theorem 6.15), $\beta^{-1}\alpha$ must be the identity transformation. That is $\alpha = \beta$.

Now given v and θv , as shown in Figure 6.22, we can find the transform θq of any point q as follows: The line $v \vee q$ meets L at a fixed point u and $\theta(v \vee q) = \theta v \vee \theta q$ passes through $\theta u = u$ and θv ; $q \vee \theta q$ passes through p (p being the centre). Hence $\theta q = (p \vee q) \wedge (\theta v \vee u)$. ◇

Theorem 6.17 *Given the stereoscope $S(D, l, r, \mu)$ of Figure 6.23, where the n^{th} image of two points, a and μa , is denoted by x^n , then \exists a central involution $\theta_n = \theta_n(l, r, D, n)$ such that $\theta_n : x^n \rightarrow x^{-n}$, $n \in \mathbb{Z}^+$.*

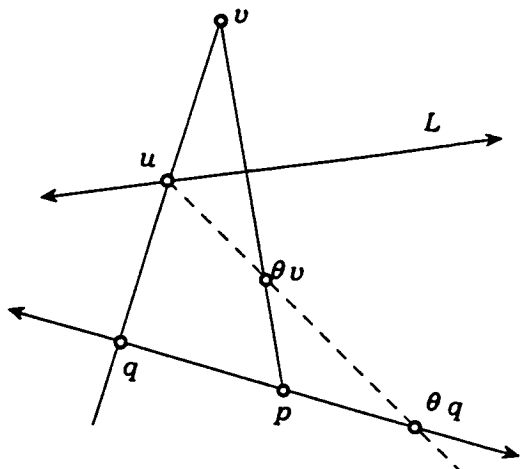


Figure 6.22: p, v and θv are collinear; p is the centre and L is the axis

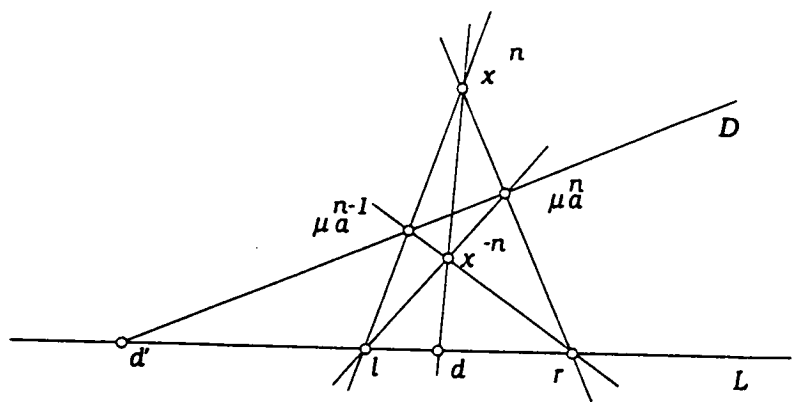


Figure 6.23: x^{-n}, x^n and d are collinear; d is the centre and D is the axis

Proof: From Theorem 6.4, x^n, x^{-n} and d are collinear. That is, $x^n, \theta_n(x^n)$ and d are collinear. Therefore by Theorem 6.16, θ_n is a (d, D) -collineation. Now if we consider Figure 6.22, and let $d = p, D = L$ and $q = l$, then we find

$$\theta_n(l) = r, \theta_n(r) = l \Rightarrow \theta_n^2(l) = l.$$

Similarly, $\theta_n^2(x) = x \forall x$ in the stereoscope. Thus θ_n is an involution. \diamond

6.2.1 Can we find a conic fixed under $\theta_n(l, r, D, n)$?

After viewing many stereograms using dual viewing, various questions arise in considering our stereoscope $S(l, D, r, \mu)$. We consider two cases in our diagrams of Figure 6.2. For example, if the set of points $\{x = (a \vee l) \wedge (\mu a \vee r) : a \in D, a \notin (l \vee r)\}$ is a planar figure of a particular type such as a conic, then is the planar figure which is made up of the homologous points $\{x^{-1} = (\mu a \vee l) \wedge (a \vee r) : a \in D, a \notin (l \vee r)\}$ also a conic?

The following result gives us an affirmative answer.

Theorem 6.18 *Any conic, C , in a plane Π is transformed into a conic by any collineation, θ .*

Proof: Suppose that the generating bases of the conic are p and q . That is, we have a projectivity $\alpha : p \bar{\wedge} q$, where p and q are points on the conic and so each will correspond to the intersection of corresponding lines in the projectivity α . Let $p = (p \vee q) \wedge Q$ where $Q \ni p$ and $q = (q \vee p) \wedge R$, where $R \in q$, then

$$\theta p = \theta(p \vee q) \wedge \theta Q, \text{ where } \theta Q \ni \theta p,$$

$$\theta q = \theta(q \vee p) \wedge \theta R, \text{ where } \theta R \ni \theta q.$$

Similarly, if we consider any line, X , of the pencil through p , then this corresponds projectively to a line $Y \ni q$ ($Y = \alpha X$) and the point $X \wedge Y$ is on the conic by definition. Now $\theta(X \wedge Y) = \theta X \wedge \theta Y$, $\theta X \ni \theta p$ and $\theta Y \ni \theta q$ which means that we have a projectivity $\theta\alpha : \theta p \bar{\wedge} \theta q$ and hence a conic. \diamond

Having established that a collineation, and in particular $\theta_n(l, r, D, n)$, will map a conic onto a conic, we can now investigate the specific question of this section : *Can we find a conic which remains fixed under $\theta_n(l, r, D, n)$?*

That is, can we find the points of a stereoscope which for dual viewing allow us to ‘see’ the same conic? It must be noted that any such conic will not contain the points l and r which cannot therefore be generating bases. Consequently, according to Lemma 6.13, the permutation $\mu : D \rightarrow D$ is not a projectivity. In order to answer our question we need to consider some important results. Consider Figure 6.24 where we have four co-planar points p_1, p_2, p_3, p_4 so that no three are collinear, which we will call a *4-pt circuit* $p_1p_2p_3p_4$. Consider the six lines joining the points taken in pairs which give

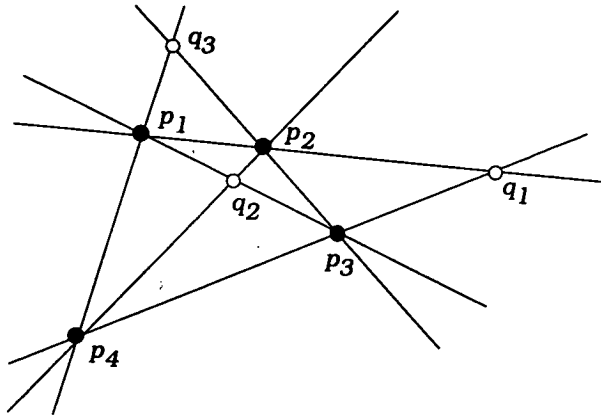


Figure 6.24: The points q_1, q_2, q_3 are diagonal points of the 4-pt circuit, $p_1p_2p_3p_4$

three more points q_1, q_2, q_3 . These three points are called the *diagonal points* of the circuit.

Theorem 6.19 *In the extended Euclidean plane the relation “being harmonic conjugate” is preserved by collineations and projectivities.*

Proof: In an extended Euclidean plane any perspectivity can be duplicated by restricting the effect of some central collineation. As every projectivity is the combination of finitely many perspectivities we need only prove this result for collineations θ . To say that c is the harmonic conjugate of d with respect to a and b means that \exists a 4-pt circuit $pqrs$. Now $\theta p\theta q\theta r\theta s$ is again a 4-pt circuit and since θ preserves incidence we have θc is the harmonic conjugate of θd with respect to θa and θb . \diamond

Lemma 6.20 *In the stereoscope $S(D, l, r, \mu)$ of Figure 6.25, the intersection of $x^n \vee d$ with D , call it d'' , is the harmonic conjugate of d with respect to x^n and x^{-n} .*

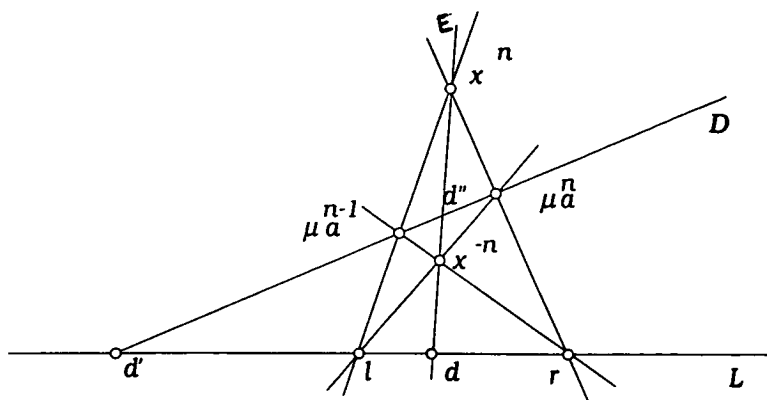


Figure 6.25: $d'' = H(d; x^n x^{-n})$

Proof: We have the following projectivity

$$L(d', l, d, r) \stackrel{x^n}{\bar{\wedge}} D(d', \mu^{n-1} a, d'', \mu^n a) \stackrel{r}{\bar{\wedge}} E(d, x^{-n}, d'', x^n)$$

and we have noted already that d is the harmonic conjugate of d' with respect to l and r . Hence by Theorem 6.19 we have the result. \diamond

Lemma 6.21 *If the vertices of a 4-pt circuit are points of a conic, the tangents at a pair of vertices meet in a point of the line joining the diagonal points of the circuit which are not on the side joining the two vertices.*

Proof: This is illustrated in Figure 6.26 and the proof may be found in Veblen and Young [29, page 115].

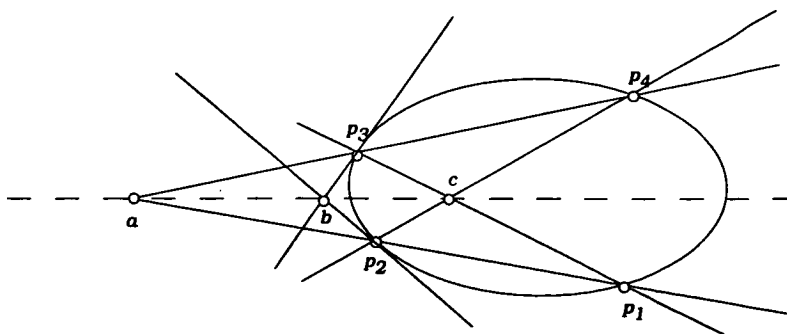


Figure 6.26: Tangents at p_2 and p_3 meet at b on $a \vee c$

Theorem 6.22 *If d is a point in the plane of a conic, but not on the conic, the points of intersection of the tangents to the conic at all the pairs of*

points, on the conic, which are collinear with d are on a line which also contains the harmonic conjugates of d with respect to these pairs of points. This is illustrated in Figure 6.27.

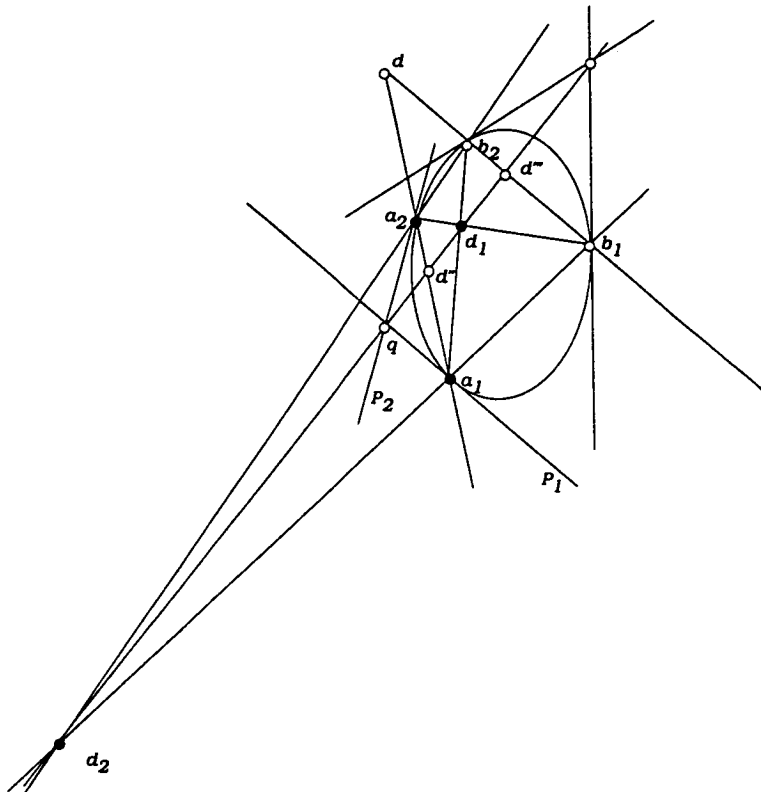


Figure 6.27: The point q , is on the line containing the harmonic conjugate d'' of d with respect to a_2 and a_1

Proof: Let a_1, a_2 and b_1, b_2 be two pairs of points on the conic which are collinear with d , and let P_1, P_2 be the tangents to the conic at a_1, a_2 respectively. If d_1, d_2 are the points $(a_1 \vee b_2) \wedge (a_2 \vee b_1)$ and $(a_1 \vee b_1) \wedge (a_2 \vee b_2)$ respectively, the line $d_1 \vee d_2$ passes through the intersection, q , of P_1 and P_2 by Lemma 6.21. Next consider the 4-pt circuit $a_2 d_2 a_1 d_1$ where d is the harmonic conjugate of $(d_2 \vee d_1) \wedge (b_2 \vee b_1) = d'''$ with respect to b_1 and b_2 . Now we have a perspectivity $(db_2 d''' b_1) \bar{\wedge} (da_2 d'' a_1)$ and if we apply Theorem 6.19 we have the point d'' in which $d_1 \vee d_2$ meets $a_1 \vee a_2$ is the harmonic conjugate of d with respect to a_1, a_2 . This shows that the line $d_1 \vee d_2 = q \vee d''$ is completely determined by the pair of points a_1, a_2 . Hence the same line $q \vee d''$ is obtained by replacing b_1, b_2 by any other pair of points on the conic collinear with d and distinct from a_1, a_2 . \diamond

Now to return to our question of a fixed conic.

Re-iterating; we have a central involution $\theta_n = \theta_n(l, r, D, n)$ such that $\theta_n : x^n \rightarrow x^{-n}$, $n \in Z^+$ which maps a conic onto a conic (Theorem 6.18). The centre, d , of this involution is the harmonic conjugate of $(l \vee r) \wedge D$ with respect to r and l . Now Theorem 6.22 tells us that if the points, x^n and $\theta_n(x^n) = x^{-n}$, collinear with d are on a conic, and we consider all such pairs, then the points of intersection of the tangents to the conic at all these pairs of points lie on a line which also contains the harmonic conjugates of d with respect to $\theta_n(x^n)$ and x^n . But by Lemma 6.20, this line of harmonic conjugates is D , the axis of our (d, D) - involution. Hence by using this information we can construct a conic, C , with the desired properties so that θ_n fixes C .

In constructing such a conic for our stereoscope we need to use the following consequence of the above results.

Lemma 6.23 *In the fixed conic C , any line joining d and the point of intersection, p_1 , of the axis, D , with the conic must be a tangent to the conic, and furthermore this point of intersection is a fixed point of θ_n , the (d, D) -involution defined on the conic.*

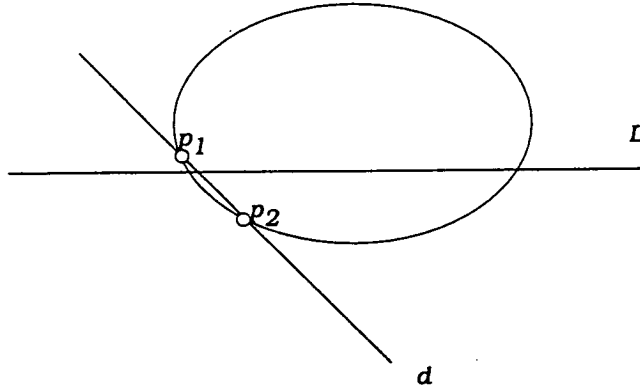


Figure 6.28: $d \vee p_1$ is a tangent and $\theta_n(p_1) = p_1$

Proof: Suppose $d \vee p_1$ cuts the conic at another point, p_2 say (see Figure 6.28), then p_1, p_2, d are collinear and by Theorem 6.22, the tangents to the conic at each of the points p_1 and p_2 intersect on D . Hence, p_2 must equal p_1 . Any point on D is a fixed point of the (d, D) -collineation by definition of a central collineation. \diamond

As mentioned previously, when the line $l \vee r$ of a stereoscope is parallel to D , then d corresponds to the midpoint of the line segment $\overline{l \vee r}$. Practically d represents m , the midpoint of the viewer's eyes. Since we normally view stereograms from the parallel position, we will construct our stereogram for the case when m represents the centre of our involution of Lemma 6.23 and the line through each row of dots, D_i , represents the axis for each i .

6.3 Constructing the Single-Image Stereogram of a Fixed Sphere

We will now test our results by constructing a Single-Image Stereogram of a fixed sphere. Each row of dots on our stereogram will represent a circular cross-section of this sphere on the sloping plane through the eyes and the row of dots. Each row of dots of the stereogram together with the eyes constitutes a stereoscope. According to our theory, the "dual" viewing of these dots, enables the viewer to perceive a circle which is an example of a fixed conic. Ideally, we would like the viewer to see as much of each circle as possible, so that a practised viewer could see almost the whole figure in the 'blink of both eyes'. That is, we see the 'inside back' of the circle with 'uncrossed' viewing and the 'outside front' of the circle with 'crossed-eye' viewing.

6.3.1 Method for finding an appropriate circle

As a consequence of Lemma 6.23, we consider the problem of finding a circle with the property that the tangent from the midpoint, m , of the viewer's eyes, touches the circle at its intersection point with the line, D_i , through a row of our stereogram.

In contrast to our earlier stereograms, our eyes will not lie on the x -axis. We let the perpendicular x and y axes pass through the D_0 , and m respectively, as shown in Figure 6.29. By D_0 , we mean the line through the central row of the stereogram which is the line of intersection of the horizontal plane through the eyes and the vertical plane through the stereogram. This vertical plane is the xz -plane. The viewer is assumed to be at a distance, d , from the stereogram, and w represents half the distance between the viewer's eyes. The co-ordinates of the intersection points of the tangents

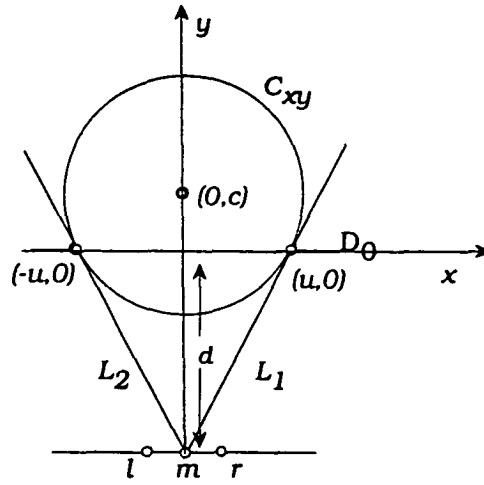


Figure 6.29: The x -axis runs through the central row of the stereogram

from $m = (0, -d)$ to the required circle are $i_1 = (u, 0)$ and $i_2 = (-u, 0)$. The other relevant co-ordinates are shown in Figure 6.29.

In order to find an appropriate circle, we can choose a sensible value of u . Sensible in the sense that if a viewer is at a distance, say 30cm, from the stereogram, then in order to perceive the correct circle clearly, and fit it on an A4 page (desirable for this thesis), then u must be approximately 6 cm for our central row of dots. Having chosen u , we can easily find the equation of the required circle by finding its centre and radius. If L_1 is the tangent through m and i_1 , then it has slope d/u and hence, the slope of the radius through the centre of the circle, $(0, c)$, is $-u/d$. Hence,

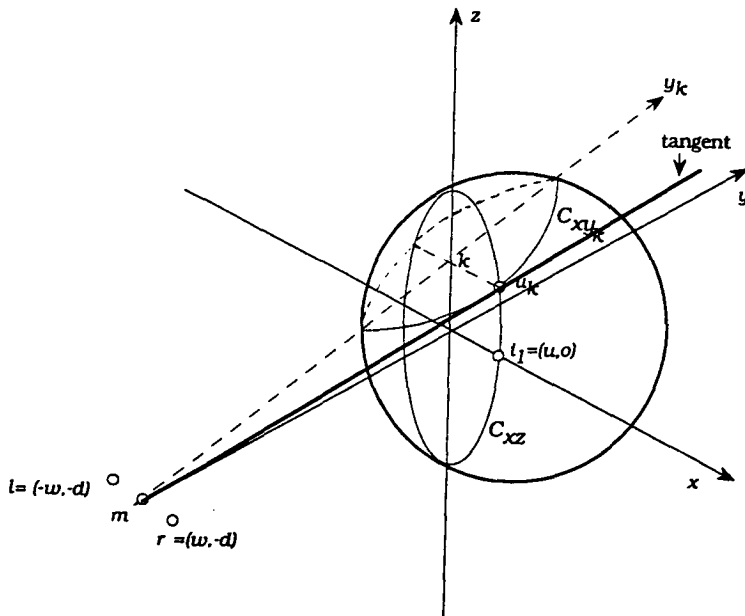
$$-c/u = -u/d \Rightarrow c = u^2/d$$

and the circle has radius $(c^2 + u^2)^{1/2}$ and its equation is

$$x^2 + (y - c)^2 = c^2 + u^2.$$

Having found the equation of the appropriate circle for the central plane section of our sphere, we can use a similar method to find the circles of intersection of each sloping plane section. The sloping plane passes through $l \vee r$ and a row of dots of the required stereogram. Once we have the appropriate circle for each plane section we can use our algorithm of equation 5.1 to construct the dots for each corresponding row of our stereogram. We now consider the problem of choosing an appropriate value of u_k for each of the sloping plane circles of intersection.

Consider the introduction of the z axis through $(0,0)$ perpendicular to the xy plane. Our stereogram will consist of rows of dots on the xz plane. The intersection of our sphere with the xz plane will give a circular cross-section, call it C_{xz} , with equation



$$x^2 + z^2 = u_k^2 \quad \text{where } z = k.$$

Solving this equation gives two values; we choose one. For any row of our stereogram of height $z = k$, the distance, d_k , to the viewer is given by $d_k = (d^2 + k^2)^{1/2}$. We could consider any sloping plane to be a rotation of our axes about the line $l \vee r$. However, for ease of calculations, we will just replace our u of subsection 6.3.1 by u_k ; and our d , by d_k . Having found the x co-ordinates of our dots, we append these with second co-ordinate k in order to plot our stereogram. The initial steps in our program of Appendix G find the circles for us. We now consider some intricacies involved in the application of our dot algorithm.

6.3.3 Boundary problems

If we create the rows of our stereogram by working from left to right, then a suitable starting dot for each row could be some arbitrarily chosen point, $(a_1, 0)$, where a_1 is close to $-u$. However, this technique is fraught with danger. To see why this is so, we need to consider what happens in a neighbourhood of $(u, 0)$. As we move along some D_k to the right we eventually

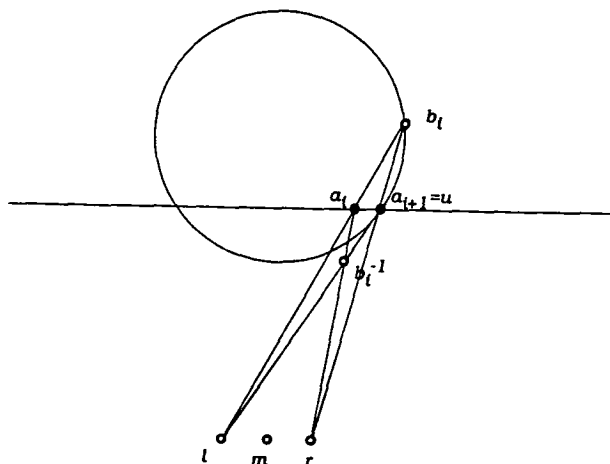


Figure 6.31: $a_{i+1} = u$

meet a point $a_i \in D_k$ which when viewed by the left eye has a corresponding point a_{i+1} for the right eye such that $a_{i+1} \geq u$. (Here when we compare the points a_{i+1} and u , we are really comparing the x co-ordinates of the points; the y co-ordinates being 0.) Figure 6.31 shows the case when $a_{i+1} = u$ and according to our geometry we would perceive the point b_i on the circle. If we then view a_{i+1} with the left eye, then only crossing our eyes would allow us to see another point of the circle. For example, the point, b_i^{-1} , if we simultaneously viewed a_i with the right eye, and if it lies on the circle. Of

course it will only lie on the circle for our special case. That is, the case when u is the tangent point on the circle for the tangent drawn from m . This is not the case in Figure 6.31 as it is drawn here.

The following question arises:

In Figure 6.32, could we see a point such as c , on the circle, where we have moved along the circle from b_i in the direction of the arrow?

We consider various scenarios.

If c is the tangent point of the line through r , then the corresponding dot is to the right of u at some point $u + \epsilon$ as shown in Figure 6.32 (a) and (b).

As the point c then moves along the circle in the direction of the arrow

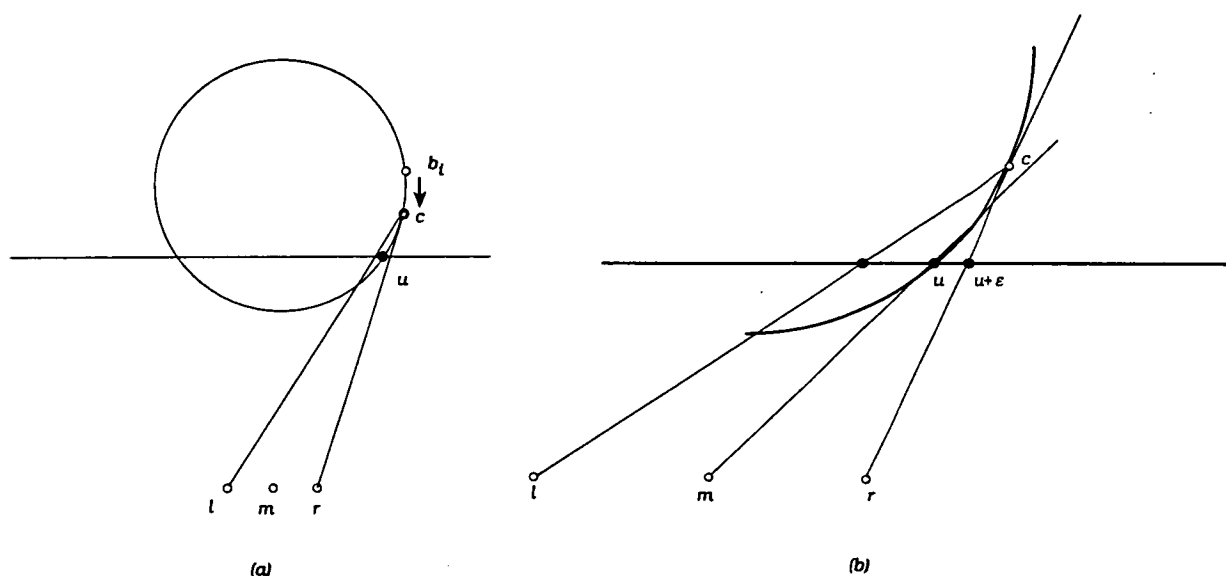


Figure 6.32: Dot for the right eye is to the right of u ; (b) An enlargement of the area of interest in (a)

we next have the situation as seen in Figure 6.33 (a) and (b), where the appropriate dot for the right eye is at a point $1 = u + \delta$ where $\delta < \epsilon$. That is, the dot for the right eye is further to the left again, and the line through r intersects the circle at two points with positive y value. Whether we see either of these points is dependent on the existence of either of the dots marked 2 and 3 in Figure 6.33 (b). We have analogous cases for 'crossed' viewing.

In reality, it seems unlikely that such intricate discussion is appropriate

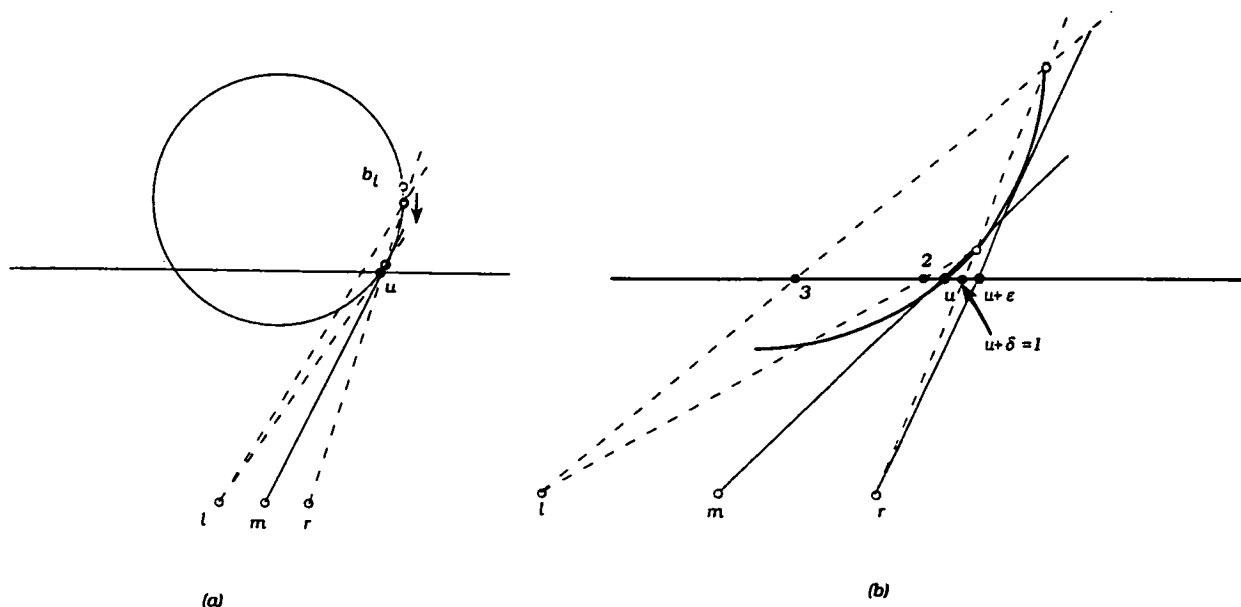


Figure 6.33: Dots change direction; (b) An enlargement of the area of interest in (a)

for our model. In order to see a point of the circle which is extremely close to the axis, using uncrossed vision, the corresponding dots would need to be so close together that we would violate the *no more than 2° disparity rule* for the single-image stereogram which has been discussed for human stereoscopic vision in section 3.1.1. Hence we must be content with seeing almost all of our circle, the ‘missing bits’ being in the vicinity of the axis as shown in Figure 6.34.

Note: Not surprisingly, in all of the currently very popular commercial single-image stereograms, the dots are spaced in such a way that the perceived three-dimensional picture does not come close to the plane of the stereogram. That is, complete figures are easily seen. For discussion of this phenomenon see section 5.1.2.

From this discussion we can see that problems begin to occur at a stage where the dot for the right eye begins to move to the left. That is, changes direction. Hence, an appropriate place to stop our process would be at a point a_i , where

$$a_{i+1} - a_i < 0. \quad (6.1)$$

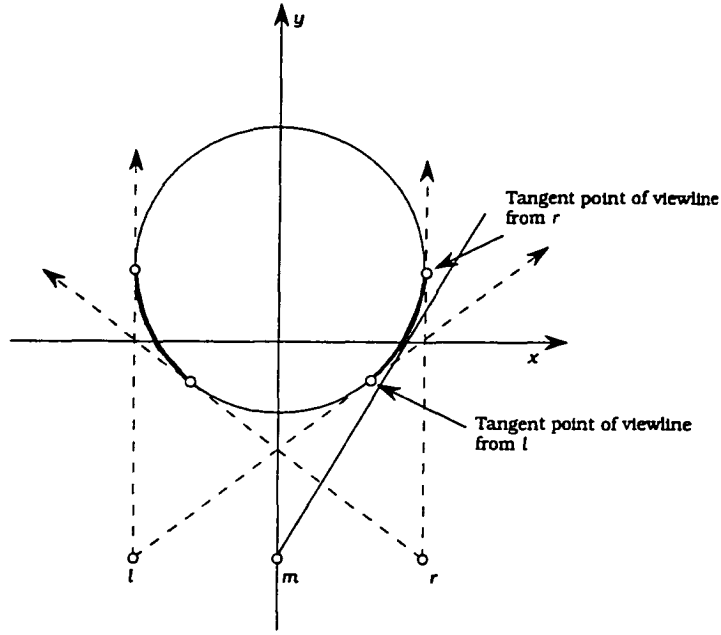


Figure 6.34: Sections of circle which may not be visible

We still need to find a suitable starting point which would alleviate this boundary problem on the left-hand side of our stereogram.

6.3.4 How do we find a suitable starting point?

Instead of employing our usual method of beginning far left, or far right, of the stereogram we will try a new approach.

Consider the line $l \vee q$ where q is the intersection point of the circle with the positive y_k -axis as shown in Figure 6.35. Suppose $p = (l \vee q) \wedge D_k$ then to choose our starting random interval we consider the following steps:-

1. Choose a set of, say 4, random x co-ordinates in the interval $(0, p)$ (which is negative) and denote this set of points (with y_k -co-ordinate 0) by **int1**. This terminology, in bold print, matches that in our program of Appendix G.
2. Apply the function of our algorithm (equation 5.1) to each point in **int1** only once. This will give us a starting interval on the positive x -axis which we denote by **int2**.
3. Apply the algorithm to **int2** which will give the row of dots of the stere-

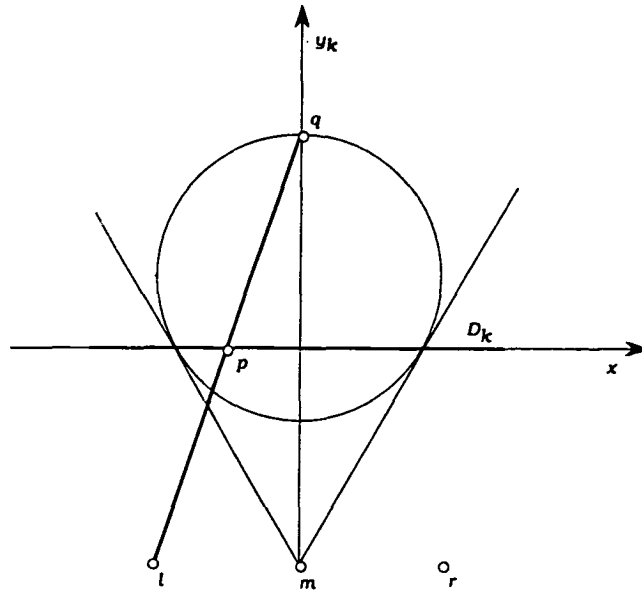


Figure 6.35: Starting point on the y_k axis for each k

ogram on the positive x -axis where we use the stopping rule of equation 6.1.

We have still to complete the row of dots by considering how to obtain the dots for the negative x -axis.

One way to execute this is to consider $-\text{lint1}$. This is the set of x -coordinates obtained by taking the negative of each co-ordinate in the set int1 .

That is:

4. Apply the algorithm to $-\text{lint1}$ which again gives a row of dots on the positive x -axis. By considering the symmetry of the situation we see that if we take the negative of this latter row of dots, we have the required row of dots on the negative x -axis. This symmetry exists because of the rather specific placement of our chosen sphere, which gives a vertical line of symmetry for the stereogram about the z -axis and for the sphere, about the yz -plane. Since this yz -plane intersects the line $l \vee r$ at m , we have a symmetrical viewing situation for each eye.

5. Now our choice of int2 allows us to join the set of dots obtained in steps 3 and 4 so that we have a complete row which when viewed with both "uncrossed" and "crossed" eyes will allow us to perceive our circle.

Note: The practised viewer can easily see this with one row of dots, but for the uninitiated, more than one row of dots is far more satisfactory.

We are now in a position to plot our stereogram where our z -values, for possible row positions, theoretically vary from $z = -u$ to $z = u$. Our program is included, and explained in detail, in Appendix G. For practical reasons, such as round-off error in calculating boundary point intersections, the range of z values is taken to be slightly smaller than the theoretical possibility.

Some resulting stereograms are shown in Figures 6.36, 6.37, 6.38 and 6.39 .

6.3.5 Comments and observations

It must be noted that in contrast to our earlier examples of Single-Image Stereograms of spheres, such as Figures 5.20 and 5.21, we have not included a background plane. In these previous cases, the inclusion of a plane was to eliminate the flat borders around the sphere as shown in Figure 5.18.

These were caused by the boundary dots being seen by one eye only. In the case of the ‘fixed sphere’, a background plane would not be fixed and so would detract from the effect of our result of Lemma 6.23.

Consequently, we must be satisfied with our circular stereogram, which could be described as containing a significant monocular cue!

The problem of the ‘flat border’ is almost eliminated by the fact that the boundary regions are very close to the page of the stereogram. This means that the dot-spacings, and hence the corresponding end intervals of dots, are very small; a dot-spacing of zero allows us to perceive a point on our page. Although our perceived image does not quite reach our page, for reasons that were discussed in subsection 6.3.3, the interval of dots seen by one eye only is still very small.

We must also note that it is desirable to have dots which are as small as possible, so that these boundary intervals of dots do not merge into solid blocks as illustrated in Figure 5.4.

These stereograms, of Figures 6.36 and 6.37, do not fit the ideal model for viewing as presented by Maeder [17, page 53], and further described in subsection 5.1.2. This ‘fixed sphere’ stereogram is unique in that it necessarily presents an image in depth which comes very near to the plane of the stereogram. That is, it approaches a depth of zero. Consequently, in order to allow the viewer to fuse the dots, care must be taken in selecting the initial

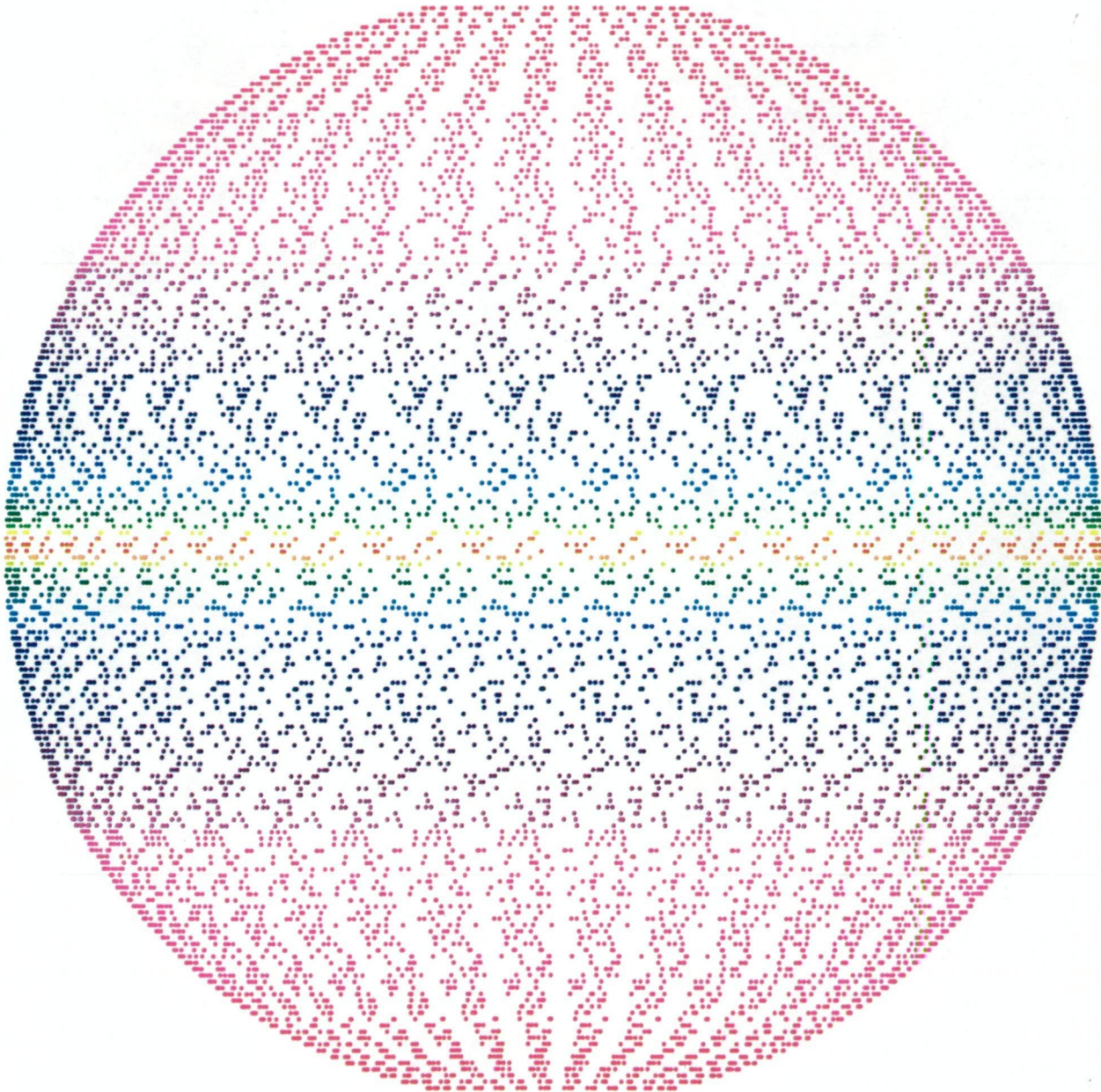


Figure 6.36: $d = 30$, $u=8$, stepsize=0.09, AbsolutePointSize=2

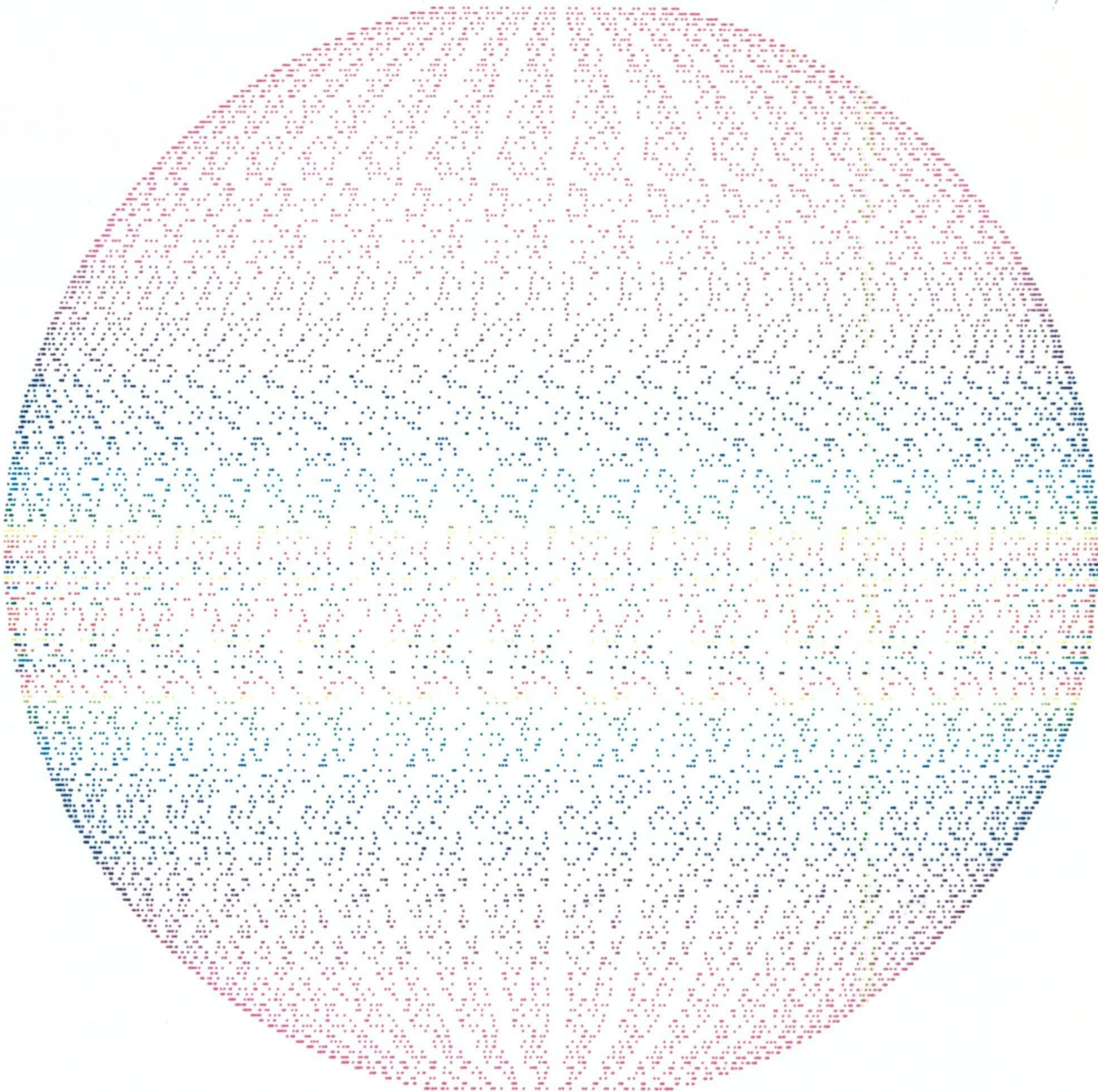


Figure 6.37: $d = 30, u=8, \text{stepsize}=0.075, \text{AbsolutePointSize}=1$

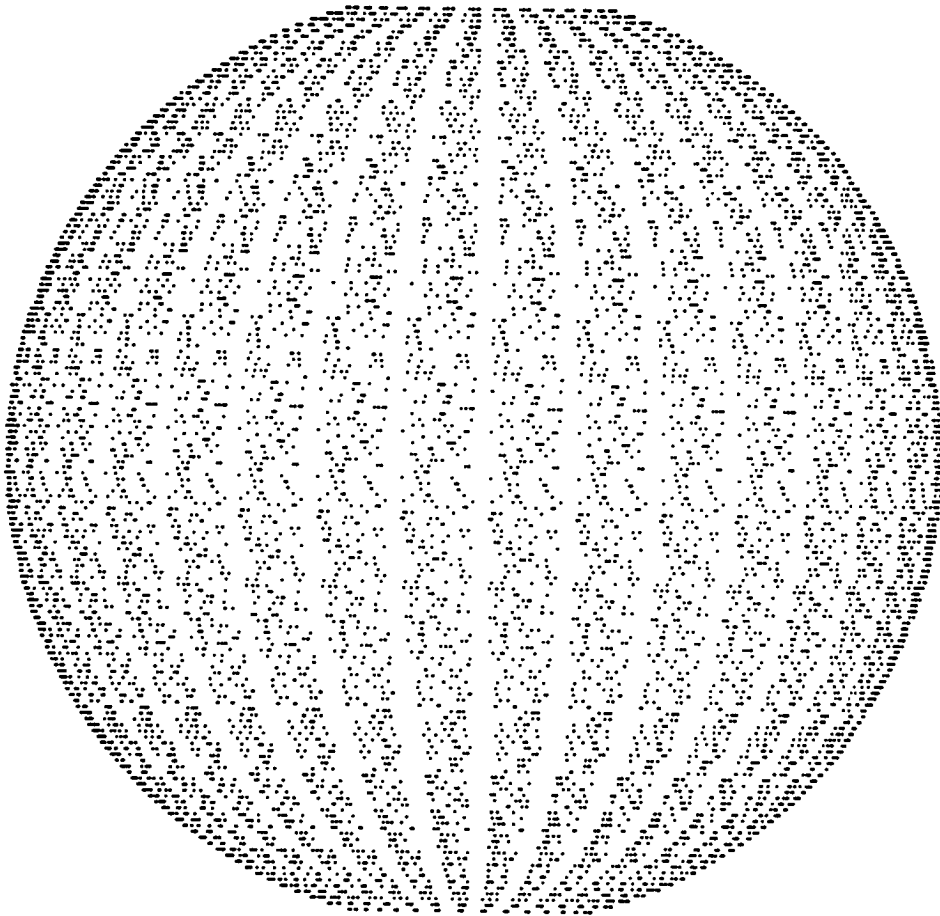


Figure 6.38: $d = 30$, $u=6$, stepsize=0.09, AbsolutePointSize=1

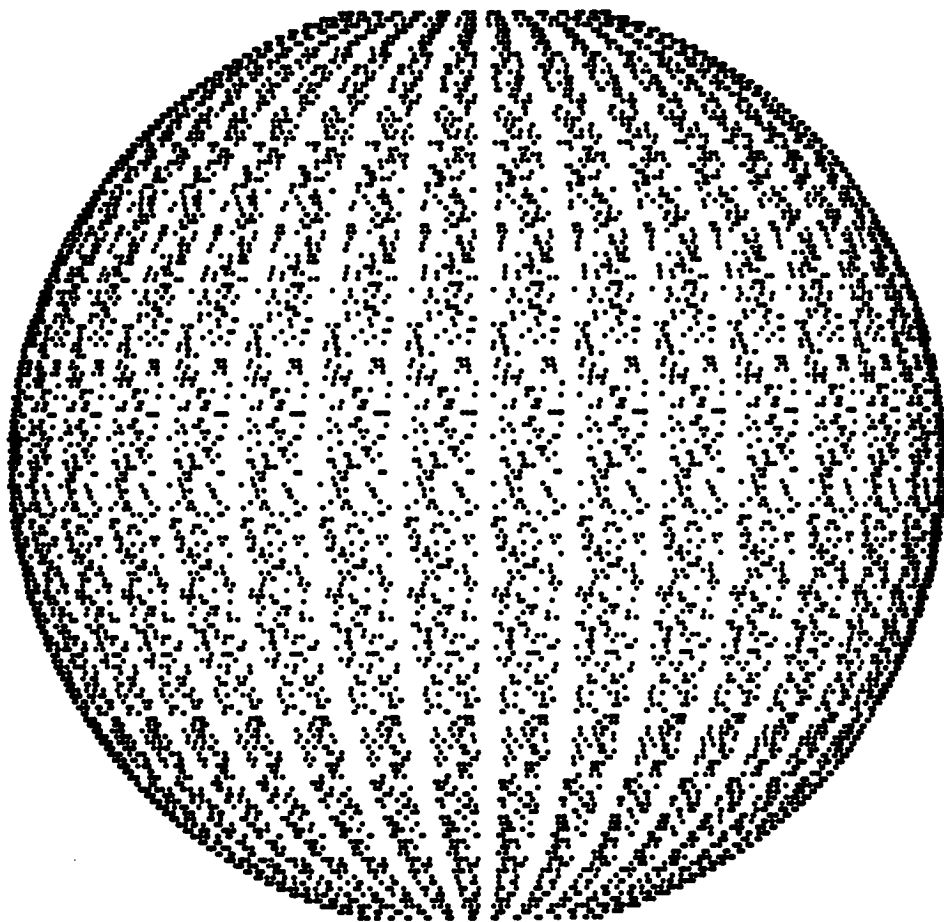


Figure 6.39: $d = 30$, $u=6$, stepsize=0.09, AbsolutePointSize=2

u of subsection 6.3.1. That is, our sphere must not be too big, so that its depth behind the plane of the stereogram relative to the distance of the viewer from the stereogram, is not too great. Further comment related to this was made in section 3.4.

It must be noted that the maximum difference in dot-spacing in Figures 6.36 and 6.37 is approximately the 1 cm suggested in section 3.4. This was measured by considering a pair of obviously matching dot clusters in the centre of our ‘sphere’, and then by considering a pairing very close to zero.

The practised viewer can easily see the fixed spheres of Figures 6.36 and 6.37 by dual viewing in the ‘blink of the eyes’. However, sadly, the ‘common’ viewer (and these seem to be in abundance!) ‘sees’ an alternative picture which is more parabolic in shape. This is not surprising, and can be easily explained by considering the discussion of alternative images in section 3.3.

In order to facilitate the viewing of the intended correct sphere, the viewing method illustrated in Figure 3.57 (b) is the ideal one. This ensures that firstly, the eyes are not ‘crossed’ too much, and secondly, that the alternating from ‘uncrossed’ to ‘crossed’ in a ‘blink of the eyes’ is easier.

Having considered both viewing techniques where the images for each eye are reversed, it seems appropriate to mention an interesting example discussed in Gregory [7, pages 126-131]. He talks about depth usually being reversed as we illustrated in Figure 6.2, but gives an example where, in practice, this is not necessarily the experience of the viewer. He provides anaglyphs of a photograph of a hollow facial mask. Examination of these examples with anaglyph ‘spectacles’ facilitates the perception of an image in depth in each of the cases; correct way around and reversed. However it is impossible to ‘see’ the nose sticking in, and not out of the face in the reverse case. He provides perceptual explanations linked with the viewer’s recognition of the photograph. That is, unlike most cases of Single-Image random dot Stereograms, the stereo pictures are monocularly recognizable.

Appendix A

Matrix Manipulations

We will briefly discuss the advantage of introducing *homogeneous* co-ordinates in order to enable us (or at least the computer) to simply calculate net transformation matrices for any composition of linear transformations such as, translating, rotating, scaling and shearing.

To begin we will work in two dimensions where any linear transformation of a point (x, y) to (T_x, T_y) can be represented by:

$$\begin{aligned}T_x &= ax + by \\T_y &= cx + dy \\or \quad \begin{bmatrix} T_x & T_y \end{bmatrix} &= \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} a & c \\ b & d \end{bmatrix}.\end{aligned}$$

In particular, we have the following 2x2 transformation matrices (for more detail see McGregor and Watt [18]).

To rotate a point (x, y) clockwise through an angle θ , about the origin as illustrated in Figure A.1, the matrix is

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

and for an anti-clockwise rotation the matrix is

$$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}.$$

For *scaling* by a factor S_x in the x-direction and S_y in the y-direction we have

$$\begin{bmatrix} T_x & T_y \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix}.$$

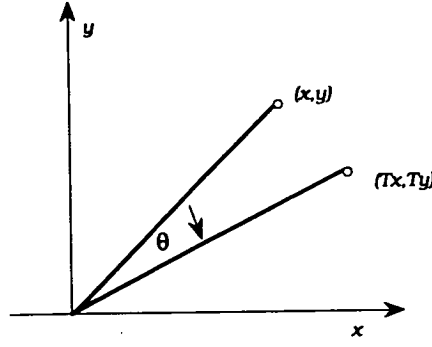


Figure A.1: Clockwise rotation

For *shearing* in the y -direction we have

$$\begin{bmatrix} T_x & T_y \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 1 & S \\ 0 & 1 \end{bmatrix},$$

and in the x -direction we have

$$\begin{bmatrix} T_x & T_y \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 1 & 0 \\ S & 1 \end{bmatrix},$$

where S is the scaling factor for the shearing.

At this stage, translations cannot be defined using a 2×2 matrix. Another drawback, is the fact that all the transformations are centered about the origin, $(0,0)$. It would be much more useful to be able to rotate or scale about an arbitrary point. McGregor and Watt [18] suggest that a homogeneous co-ordinate system overcomes these difficulties.

Homogeneous co-ordinates are a system of co-ordinates developed so that analytic methods could be applied to the Extended Euclidean plane of Projective geometry. A description of their determination can be found in Courant and Robbins [4, page193].

In such a system, a point (x, y) becomes (rx, ry, r) .

To show how these homogeneous co-ordinates help us to obtain a translation matrix, we consider the following:

Suppose our translation in terms of our original co-ordinates is

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

That is, our translation in the x -direction is α and our translation in the y -direction is β . To express this in terms of the homogeneous co-ordinates, (X, Y, Z) , and a matrix, A say, then

$$\begin{bmatrix} X' & Y' & Z' \end{bmatrix} = \begin{bmatrix} X & Y & Z \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix},$$

where $x = X/Z$, $y = Y/Z$, $x' = X'/Z'$ and $y' = Y'/Z'$.

After multiplying these matrices, and solving for the matrix elements, a_{ij} , $1 \leq i, j \leq 3$, we find that

$$\begin{bmatrix} A \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \alpha & \beta & 1 \end{bmatrix},$$

is the required translation matrix.

For convenience, let $r = 1$ and then the point becomes $(x, y, 1)$. Transformation matrices are now 3×3 and we have the advantage that we can represent a translation by a 3×3 matrix. The most common transformation matrices become:

For translation T_x in the x direction and T_y in the y direction

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ T_x & T_y & 1 \end{bmatrix},$$

for rotation (clockwise) about origin

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

for rotation (anti-clockwise) about origin

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

for scaling of S_x in the x direction and S_y in the y direction

$$\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and for shearing in the x -direction

$$\begin{bmatrix} 1 & 0 & 0 \\ S & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Figure A.2 (a) shows a square of sidelength 2 cm with vertex at the origin of the rectangular co-ordinate system. Each of the above transformations are applied to this square to give new figures. The square of Figure A.2 (b) is obtained by translating the square of (a) by a distance of 1 cm in each of the x and y directions. The figure of Figure A.2 (c) is obtained by rotating the

square of (a) in a clockwise direction about the origin by an angle of $\pi/4$, while the figure of Figure A.2 (d) is the result of an anti-clockwise rotation of $\pi/6$ about the origin. Scaling by a factor of 1.5 in the x direction and 1.25 in the y direction gives the rectangle of Figure A.2 (e), while shearing by a factor of 1 in the x and y directions gives the figures of Figures A.2 (f) and (g) respectively.

Suppose we want to rotate a rectangle, in any position, by an angle of ϕ anti-clockwise about its bottom left-hand vertex, $p = (x, y)$. This can easily be done now using the following steps.

Firstly we translate by the matrix, T_1 so that the vertex, p , is at the origin where

$$T_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -x & -y & 1 \end{bmatrix}.$$

Next we rotate about the origin using the rotation matrix R where

$$R = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Lastly, we translate the rotated rectangle so that p is back to its original position using the translation matrix T_2 , where

$$T_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x & y & 1 \end{bmatrix}.$$

The net transformation matrix becomes T_1RT_2 where

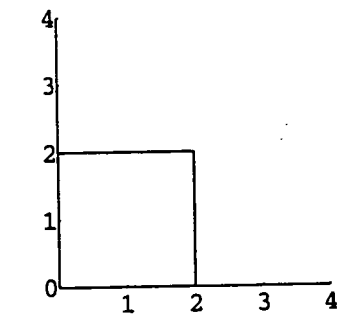
$$T_1RT_2 = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ x(1 - \cos\phi) + y\sin\phi & y(1 - \cos\phi) + x\sin\phi & 1 \end{bmatrix}$$

Application of this matrix to a square of sidelength 2 cm with its bottom left-hand vertex at the point (1, 1.5), and rotated by $\pi/5$ in an anti-clockwise direction, gives the square of Figure A.3. A net transformation matrix is always of the form

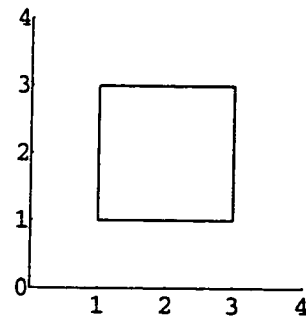
$$\begin{bmatrix} a & d & 0 \\ b & e & 0 \\ c & f & 1 \end{bmatrix},$$

and so the multiplication

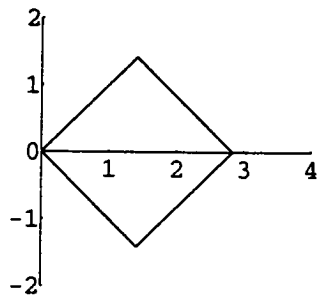
$$\begin{bmatrix} T_x & T_y & 1 \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} a & d & 0 \\ b & e & 0 \\ c & f & 1 \end{bmatrix}$$



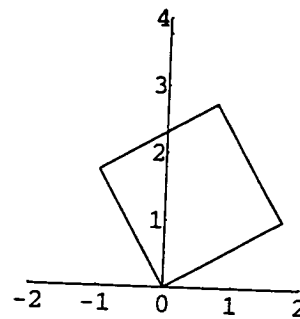
(a)



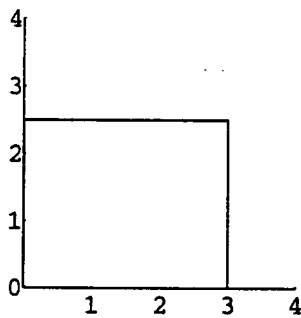
(b)



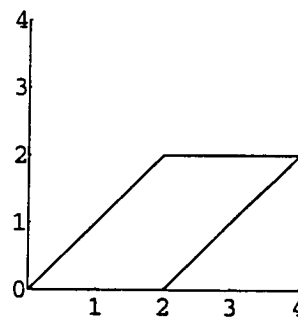
(c)



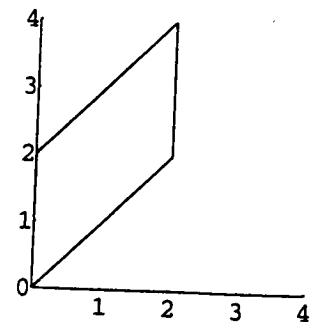
(d)



(e)



(f)



(g)

Figure A.2: Square; translation, clockwise rotation, anti-clockwise rotation, scaling, xshear, yshear

Consider a square with bottom left-hand corner placed at the origin of our rectangular co-ordinate system. A parametric representation of this square is given by:

```
Squaredges[dim_,t_] := {{0,dim}+t*({dim,dim}-{0,dim}),
{dim,dim}+t*({dim,0}-{dim,dim}),
{dim,0}+t*({0,0}-{dim,0}),{0,0}+t*({0,dim}-{0,0})}
```

Translation matrix using homogeneous co-ordinates:

```
transl2[Tx_,Ty_] := {{1,0,0},{0,1,0},{Tx,Ty,1}}
```

Rotation matrix for a rotation of *ang* about the origin in a clockwise direction:

```
rotc[ang_] := {{Cos[ang],-Sin[ang],0},{Sin[ang],Cos[ang],0},
{0,0,1}}
```

Rotation matrix for a rotation of *ang* about the origin in an anti-clockwise clockwise direction:

```
rotac[ang_] := {{Cos[ang],Sin[ang],0},{-Sin[ang],Cos[ang],0},
{0,0,1}}
```

Scaling matrix for scaling by a factor of *Sx* in the x direction and *Sy* in the y direction:

```
scale2[Sx_,Sy_] := {{Sx,0,0},{0,Sy,0},{0,0,1}}
```

Matrix for shearing in the x direction:

```
shearx[S_] := {{1,0,0},{S,1,0},{0,0,1}}
```

Matrix for shearing in the y direction:

```
sheary[S_] := {{1,S,0},{0,1,0},{0,0,1}}
```

The following products give the points on the new figure which results from applying the previous transformations to the original square (the function names are self-explanatory):

```
squtrans[dim_,t_,Tx_,Ty_] := Map[Drop[#, -1]&, Map[Insert[#, 1, 3]&,
Squaredges[dim,t]].transl2[Tx,Ty]]
squrotc[dim_,t_,ang1_] := Map[Drop[#, -1]&, Map[Insert[#, 1, 3]&,
Squaredges[dim,t]].rotc[ang1]]
squrotac[dim_,t_,ang2_] := Map[Drop[#, -1]&, Map[Insert[#, 1, 3]&,
Squaredges[dim,t]].rotac[ang2]]
squscale[dim_,t_,Sx_,Sy_] := Map[Drop[#, -1]&, Map[Insert[#, 1, 3]&,
Squaredges[dim,t]].scale2[Sx,Sy]]
squshearx[dim_,t_,S_] := Map[Drop[#, -1]&, Map[Insert[#, 1, 3]&,
Squaredges[dim,t]].shearx[S]]
squsheary[dim_,t_,S_] := Map[Drop[#, -1]&, Map[Insert[#, 1, 3]&,
Squaredges[dim,t]].sheary[S]]
```

The following expressions allow us to plot the resulting figures for the particular cases given in the individual arguments of the transformation functions:

```

ParametricPlot[Evaluate[Squaredges[2,t]],{t,0,1},
PlotRange->{{0,4},{0,4}},AspectRatio->Automatic,
Ticks->{{0,1,2,3,4},{0,1,2,3,4}}]

ParametricPlot[Evaluate[squtrans[2,t,1,1]],{t,0,1},
PlotRange->{{0,4},{0,4}},AspectRatio->Automatic,
Ticks->{{0,1,2,3,4},{0,1,2,3,4}}]

ParametricPlot[Evaluate[squrotc[2,t,Pi/4]],{t,0,1},
PlotRange->{{0,4},{-2,2}},AspectRatio->Automatic,
Ticks->{{0,1,2,3,4},{-2,-1,0,1,2}}]

ParametricPlot[Evaluate[squrotac[2,t,Pi/6]],{t,0,1},
PlotRange->{{-2,2},{0,4}},AspectRatio->Automatic,
Ticks->{{-2,-1,0,1,2},{0,1,2,3,4}}]

ParametricPlot[Evaluate[squrscale[2,t,1.5,1.25]],{t,0,1},
PlotRange->{{0,4},{0,4}},AspectRatio->Automatic,
Ticks->{{0,1,2,3,4},{0,1,2,3,4}}]

ParametricPlot[Evaluate[squshearx[2,t,1]],{t,0,1},
PlotRange->{{0,4},{0,4}},AspectRatio->Automatic,
Ticks->{{0,1,2,3,4},{0,1,2,3,4}}]

ParametricPlot[Evaluate[squsheary[2,t,1]],{t,0,1},
PlotRange->{{0,4},{0,4}},AspectRatio->Automatic,
Ticks->{{0,1,2,3,4},{0,1,2,3,4}}]

```

Now for a square with bottom left-hand co-ordinates (a,b)

```

Squaredges2[a_,b_,dim_,t_] := ({a,b}+t*({a,b+dim}-{a,b}),
{a,b+dim}+t*({a+dim,b+dim}-{a,b+dim}),
{a+dim,b+dim}+t*({a+dim,b}-{a+dim,b+dim}),
{a+dim,b}+t*({a,b}-{a+dim,b}))

```

If we want to rotate this square about the point (a,b), then firstly we need to translate it back to the origin ; we then rotate it about the origin using our rotation matrix and then we translate the result back to the original position of the corner (a,b).

Suppose (a,b) = (1,1.5) and that we want to rotate anti-clockwise about this point by an angle of $\pi/5$ then we consider the following matrix product:

```

transform[a_,b_,ang2_] := transl2[-a,-b].rotac[ang2].transl2[a,b]

squtransform[a_,b_,dim_,t_,ang2_] := Map[Drop[#, -1]&, Map[Insert[#, 1, 3]&
Squaredges2[a,b,dim,t]].transform[a,b,ang2]]

ParametricPlot[Evaluate[squtransform[1,1.5,2,t,Pi/5]],{t,0,1},
PlotRange->{{-1,3},{1,5}},AspectRatio->Automatic,
Ticks->{{-1,0,1,2,3},{1,2,3,4,5}}]

```

Appendix B

Program for creating perspective drawings

□ This program

(i) Gives us the co-ordinates of the vertices of a cube obtained by applying any composition of linear transformations to a cube originally placed about the origin in general position shown in Figure (3.10) Chapter 3.

(ii) Plots a perspective drawing of the cube or a rectangular grid for given viewpoint and drawing surface. In particular, for a drawing surface of a plane, a circular cylinder, a sphere and a cone. The particular surface equation is given by the function, ObjectSurface.

□ Note: Each time the program is run we need to enter the ObjectSurface and the ViewPt first and if we want the furtherest picture for each surface we must amend PictureList with Min replaced by Max. Also if we are viewing along the z axis in Objectcube we must have Delete[#,3] i.e. 3 replaces 2

■ Cylinder

```
ObjectSurface[xpt_List]:=xpt[[1]]^2+xpt[[2]]^2-9;
```

■ Cone

```
ObjectSurface[xpt_List]:=xpt[[1]]^2+xpt[[2]]^2-1/3*xpt[[3]]^2;
```

■ Plane

```
ObjectSurface[xpt_List]:=xpt[[2]]
```

■ Sphere

```
ObjectSurface[xpt_List]:=xpt[[1]]^2+xpt[[2]]^2+xpt[[3]]^2-16;
```

```
MakeGrid2[{ycentre_,zcentre_},{dy_,dz_},{ny_,nz_},t_]:=
Join[Table[{ny*dy*t+ycentre,i*dz+zcentre},{i,-nz,nz}],
Table[{i*dy+ycentre,nz*dz*t+zcentre},{i,-ny,ny}]];
```

The following function ImageGrid gives the parametric representation of a rectangular grid.

```
ImageGrid[t_]:=MakeGrid2[{0,5},{1,1},{4,4},t]
```

Note: In order to have the correct scale, editstyles must be set for each individual example. To set correctly, plot each with axes first to see the maximum and minimum x and y values , particularly for the stretched case.

Next we plot **ImageGrid**

```
ParametricPlot[
  Evaluate[ImageGrid[t]], {t, -1, 1}, Axes->True,
  PlotStyle->AbsoluteThickness[1],
  PlotRange->{{-10, 10}, {0, 20}}, AspectRatio->Automatic]
```

Image1 gives the grid on the xz-plane or y=0.

Image2 gives the grid on the xy-plane or z=0.

```
Image[t_] := Map[transz, Map[Insert[#, 0, 2] &, ImageGrid[t]]]
```

```
Image1[t_] := Map[Insert[#, 0, 2] &, ImageGrid[t]]
```

Image1h gives the homogeneous co-ordinates of the grid on the xz-plane or y=0.

Image2h gives the homogeneous co-ordinates of the grid on the xy-plane or z=0.

```
Image1h[t_] := Map[Insert[#, 1, 4] &, Image1[t]]
```

```
Image2[t_] := Map[Insert[#, 0, 3] &, ImageGrid[t]]
```

```
Image2h[t_] := Map[Insert[#, 1, 4] &, Image2[t]]
```

GridImage gives any transformation of the grid multiplying by matrix of homogeneous co-ordinates. This results in a set of points with three co-ordinates only as we saw in Appendix A.

```
GridImage[scalex_, scaley_, scalez_, Tx_, Ty_, Tz_, ang1_, ang2_,
  ang3_, t_] := Image2h[t].nettransf[scalex, scaley, scalez, Tx, Ty, Tz, ang1, ang2, ang3]
```

rh and **uh** define rotation matrices using homogeneous co-ordinates as detailed in Appendix A. In this case **rh** is the composition of a rotation by **ang3** about firstly the z axis and then about the y axis. **uh** is the composition of a rotation by **ang2** about the x axis and then by an angle of **ang1** about the y axis.

```
rh[ang3_] := {{Cos[ang3], Sin[ang3], 0, 0}, {-Sin[ang3], Cos[ang3], 0, 0},
  {0, 0, 1, 0}, {0, 0, 0, 1}}.
{{Cos[ang3], 0, Sin[ang3], 0}, {0, 1, 0, 0}, {-Sin[ang3], 0, Cos[ang3], 0},
  {0, 0, 0, 1}};
```

```
uh[ang2_, ang1_] := {{1, 0, 0, 0}, {0, Cos[ang2], Sin[ang2], 0},
  {0, -Sin[ang2], Cos[ang2], 0}, {0, 0, 0, 1}}.
{{Cos[ang1], 0, Sin[ang1], 0}, {0, 1, 0, 0}, {-Sin[ang1], 0, Cos[ang1], 0},
  {0, 0, 0, 1}};
```

scale is a scaling matrix for scaling by a factor of **scalei** in the i-direction.

```
scale[scalex_, scaley_, scalez_] :=
  {{scalex, 0, 0, 0}, {0, scaley, 0, 0}, {0, 0, scalez, 0}, {0, 0, 0, 1}};
```

transl is a translation matrix for translating by **Tj** units in the jth direction.

```
transl[Tx_, Ty_, Tz_] :=
  {{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {Tx, Ty, Tz, 1}};
```

nettransf is the net transformation matrix.


```

netransf[scalex_,scaley_,scalez_,Tx_,Ty_,Tz_,ang1_,ang2_,ang3_] :=
Map[
Drop[#, -1]&, rh[ang3].uh[ang2, ang1].scale[scalex, scaley, scalez].
transl[Tx, Ty, Tz]];

```

corners gives the co-ordinates of the cube vertices once the cube, originally in general position about the origin, has been transformed by particular cases of the preceding transformations.

```

corners[scalex_,scaley_,scalez_,Tx_,Ty_,Tz_,ang1_,ang2_,ang3_,dim_] :=
Flatten[Table[{{-dim,j,-dim,1}.
netransf[scalex,scaley,scalez,Tx,Ty,Tz,ang1,ang2,ang3],
{-dim,j,dim,1}.
netransf[scalex,scaley,scalez,Tx,Ty,Tz,ang1,ang2,ang3],
{dim,j,dim,1}.netransf[scalex,scaley,scalez,Tx,Ty,Tz,ang1,ang2,ang3],
{dim,j,-dim,1}.netransf[scalex,scaley,scalez,Tx,Ty,Tz,ang1,ang2,ang3]
{j,-dim,dim,2*dim}]/N,1];

```

cubedges gives a list of the parametric representations of the lines containing the edges of the cube. It must be noted that the order of the vertices is crucial here. The order for which this program is written is shown in Figure ? To obtain the correct line segments to represent the edges of the cube, we consider values of the parameter, t , between 0 and 1.

```

cubedges[scalex_,scaley_,scalez_,Tx_,Ty_,Tz_,ang1_,ang2_,
ang3_,dim_,t_] :=
(c=corners[scalex,scaley,scalez,Tx,Ty,Tz,ang1,ang2,ang3,dim];
{c[[1]]+t*(c[[2]]-c[[1]]),
c[[2]]+t*(c[[6]]-c[[2]]),
c[[7]]+t*(c[[3]]-c[[7]]),
c[[4]]+t*(c[[3]]-c[[4]]),
c[[2]]+t*(c[[3]]-c[[2]]),
c[[1]]+t*(c[[4]]-c[[1]]),
c[[1]]+t*(c[[5]]-c[[1]]),
c[[5]]+t*(c[[8]]-c[[5]]),
c[[4]]+t*(c[[8]]-c[[4]]),
c[[5]]+t*(c[[6]]-c[[5]]),
c[[8]]+t*(c[[7]]-c[[8]]),
c[[6]]+t*(c[[7]]-c[[6]])});

```

Viewline gives the parametric representation of the line through the viewpoint, represented by its three co-ordinates, and some point represented by its list of three co-ordinates, **xpt_List**.

```

ViewLine[ViewPt_,xpt_List,t_] := ViewPt + t*(xpt - ViewPt);
SurfaceIntersectionList[ViewPt_,xpt_List] := zt /.
Solve[ObjectSurface[ViewLine[ViewPt,xpt,zt]] == 0, zt];

```

PictureList represents the value of the parameter, t , at the intersection point of the viewline with the picture surface. In the following case, the minimum such value of t is returned. In cases where the viewline cuts the picture surface twice, the other value may be found by replacing the **Min** by **Max**.

```
PictureList[ViewPt_, xpt_List] := Map[ViewLine[ViewPt, xpt, #] &,
List[Min[SurfaceIntersectionList[ViewPt, xpt]]]];
```

For all real values of t between 0 and 1, **Objectcube**, represents all the points on the perspective drawing of the edges of our cube (that is, the perspective drawing of the skeleton cube). These are points on our chosen picture surface.

```
Objectcube[ViewPt_, scalex_, scaley_, scalez_, Tx_, Ty_, Tz_, ang1_, ang2_,
ang3_, dim_, t_] :=
Map[Delete[#, 2] &,
Flatten[Map[PictureList[ViewPt, #] &,
cubedges[scalex, scaley, scalez, Tx, Ty, Tz, ang1, ang2, ang3, dim, t]], 1]];
```

Objectgrid gives the parametric representation of the perspective drawing of the grid on our picture surface.

```
Objectgrid[ViewPt_, scalex_, scaley_, scalez_, Tx_, Ty_, Tz_, ang1_, ang2_,
ang3_, t_] :=
Map[Delete[#, 2] &,
Flatten[Map[PictureList[ViewPt, #] &,
GridImage[scalex, scaley, scalez, Tx, Ty, Tz, ang1, ang2, ang3, t]], 1]];
```

PerspectiveDrawinggrid gives the parametric plot of the perspective drawing of the grid on our picture surface. The parameter takes all real values between -1 and 1. To be seen correctly it must be wrapped around the appropriate picture surface, if this is curved.

```
PerspectiveDrawinggrid[ViewPt_, scalex_, scaley_, scalez_, Tx_, Ty_, Tz_,
ang1_, ang2_, ang3_] :=
ParametricPlot[Evaluate[
Objectgrid[ViewPt, scalex, scaley, scalez, Tx, Ty, Tz, ang1, ang2, ang3, cat]],
{cat, -1, 1},
PlotRange->{{-8, 8}, {0, 16}},
PlotStyle->AbsoluteThickness[1], AspectRatio->Automatic];

PerspectiveDrawinggrid[{2, -20, 30}, 1, 1, 1, 0, 0, 0, 0, 0, 0]
```

PerspectiveDrawingcube gives the parametric plot of the perspective drawing of the skeleton cube on our picture surface. The parameter takes all real values between 0 and 1. To be seen correctly it must be wrapped around the appropriate picture surface, if this is curved.

```
PerspectiveDrawingcube[ViewPt_, scalex_, scaley_, scalez_, Tx_, Ty_, Tz_,
ang1_, ang2_, ang3_, dim_] :=
ParametricPlot[Evaluate[
Objectcube[ViewPt, scalex, scaley, scalez, Tx, Ty, Tz, ang1, ang2, ang3,
dim, cat]], {cat, 0, 1},
PlotRange->{{-8, 8}, {0, 16}},
PlotStyle->AbsoluteThickness[1], AspectRatio->Automatic];

PerspectiveDrawingcube[{0, -30, 22}, 1, 1, 1, 0, 0, 4, 0, Pi/6, Pi/4, 1.5]
```

Appendix C

Program for Lunenburg Experiment

We begin by finding the centre and radius of the Vieth-Muller circle for a given fixation point, $(p,q,0)$. This circle also passes through the nodal points of the viewer's eyes, which we have placed at $(-2.875,0,0)$ and $(2.875,0,0)$. We have three points of a circle, and so to find its centre, we need the circumcentre of the triangle with these three points as vertices. In this program, we have used a fixation point of $(0,15,0)$, to give our **ObjectSurface** function which represents a cylinder with horizontal circular cross-section equal to the Vieth-Muller circle for this fixation point.

Note: If this is to mimick Luneburg's experiment, we must ensure that we have a Vieth-Muller circle which goes through the intersection points of the corresponding rays of our finite sub-pencils. This is a special case where we are considering equal angles between neighbouring rays of each sub-pencil. For this example 'Geometer Sketchpad' was used to make a scale drawing of an appropriate pair of sub-pencils (their spacing equals eye-separation) and the set angle was chosen to be $\pi/10$. Using the method described above we could then find the radius, and centre, of an appropriate Vieth-Muller circle.

```
ObjectSurface[xpt_List] := (xpt[[1]])^2 +
(xpt[[2]] - 9.53)^2 - (9.96)^2;
```

lines returns the parametric equation of an arbitrarily chosen line; through the points $(0,0,0)$ and $(-11,11,0)$.

```
lines[t_] := {(0, 0, 0) + t * {-11, 11, 0}}
```

transl is a translation matrix using homogeneous co-ordinates.

```
transl[Tx_, Ty_, Tz_] :=
{{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {Tx, Ty, Tz, 1}};
```

uh is a rotation matrix about the origin.

```
uh[ang1_] := {{Cos[ang1], -Sin[ang1], 0, 0}, {Sin[ang1], Cos[ang1], 0, 0},
{0, 0, 1, 0}, {0, 0, 0, 1}};
```

linesi, where i is integral and $2 \leq i \leq 6$ represent lines which are rotations (in multiples of $\pi/8$), and then translations, of **linest**. They are concurrent at the nodal point of the right eye.

linesil, where i is integral and $2 \leq i \leq 6$ represent lines which are rotations (in multiples of $\pi/8$), and then translations, of **linest**. They are concurrent at the nodal point of the left eye.

```
lines2[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . transl[2.875, 0, 0], {2}]
lines2l[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . transl[-2.875, 0, 0], {2}]
lines3[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[Pi/10] . transl[2.875, 0, 0], {2}]
lines3l[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[Pi/10] . transl[-2.875, 0, 0], {2}]
lines4[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[Pi/5] . transl[2.875, 0, 0], {2}]
lines4l[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[Pi/5] . transl[-2.875, 0, 0], {2}]
lines5[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[3 Pi/10] . transl[2.875, 0, 0], {2}]
lines5l[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[3 Pi/10] . transl[-2.875, 0, 0], {2}]
lines6[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[4 Pi/10] . transl[2.875, 0, 0], {2}]
lines6l[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]] . uh[4 Pi/10] . transl[-2.875, 0, 0], {2}]
```

```

lines7[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]].uh[5 Pi/10].t;
lines7l[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]].uh[5 Pi/10].t;
lines8[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]].uh[6 Pi/10].t;
lines8l[t_] := Map[Drop[#, -1] &, Map[Append[#, 1] &, lines[t]].uh[6 Pi/10].t;

```

parametersi and parametersil return the values of the parameters for each of the intersection points, of the linesi or linesil, with the Vieth-Muller circle.

```

parameters1=zt /.
Solve[ObjectSurface[lines2[zt][[1]]]==0,zt]

parameters1l=zt /.
Solve[ObjectSurface[lines2l[zt][[1]]]==0,zt]

parameters2=zt /.
Solve[ObjectSurface[lines3[zt][[1]]]==0,zt]

parameters2l=zt /.
Solve[ObjectSurface[lines3l[zt][[1]]]==0,zt]

parameters3=zt /.
Solve[ObjectSurface[lines4[zt][[1]]]==0,zt]

parameters3l=zt /.
Solve[ObjectSurface[lines4l[zt][[1]]]==0,zt]

parameters4=zt /.
Solve[ObjectSurface[lines5[zt][[1]]]==0,zt]

parameters4l=zt /.
Solve[ObjectSurface[lines5l[zt][[1]]]==0,zt]

parameters5=zt /.
Solve[ObjectSurface[lines6[zt][[1]]]==0,zt]

parameters5l=zt /.
Solve[ObjectSurface[lines6l[zt][[1]]]==0,zt]

parameters6=zt /.
Solve[ObjectSurface[lines7[zt][[1]]]==0,zt]

parameters6l=zt /.
Solve[ObjectSurface[lines7l[zt][[1]]]==0,zt]

parameters7=zt /.
Solve[ObjectSurface[lines8[zt][[1]]]==0,zt]

parameters7l=zt /.
Solve[ObjectSurface[lines8l[zt][[1]]]==0,zt]

```

Next we plot each line segment, of the original linesi or linesil, from the nodal point to the Vieth-Muller circle. We use the values of the parameters returned above.

```

ParametricPlot[Evaluate[
Map[Delete[#, 3] &, linesi[cat]],
{cat, 0, 0.918134}, Axes->True, PlotStyle->RGBColor[0, 1, 0],
PlotRange->{{-11, 11}, {0, 20}}, AspectRatio->Automatic]

```

```

ParametricPlot[Evaluate[
Map[Delete[#, 3]&, lines1[cat]]],
{cat, 0, 0.395407}, Axes->True, PlotStyle->RGBColor[1, 0, 0],
PlotRange->{{-11, 11}, {0, 20}}, AspectRatio->Automatic]

```

Next we find the perspective drawing of these pencils, with **ObjectSurface** being the cylinder. We need the perspective drawing of the pencil with vertex $(2.875, 0, 0)$ from a viewpoint directly above this vertex; from a point such as $(2.875, 0, 10)$. Similarly, we need the perspective drawing of the pencil with vertex $(-2.875, 0, 0)$, from a viewpoint of $(-2.875, 0, 10)$. This mimicks Luneburg's experiment.

Note: In **Objectlines** we need to insert the appropriate **lines1**.

In **PerspectiveDrawing** we then need to insert the correct values for the parameter. Since we can't see the vertex if we are directly above it, we begin by selecting a parameter value which is bigger than 0. This program needs some work to make it more automatic.

```

ViewLine[ViewPt_, xpt_List, t_] := ViewPt + t*(xpt - ViewPt);
SurfaceIntersectionList[ViewPt_, xpt_List] := zt /.
Solve[ObjectSurface[ViewLine[ViewPt, xpt, zt]] == 0, zt];

PictureList[ViewPt_, xpt_List] := Map[ViewLine[ViewPt, xpt, #]&,
List[Max[SurfaceIntersectionList[ViewPt, xpt]]]];

Objectlines[ViewPt_, t_] :=
Map[Delete[#, 2]&,
Flatten[Map[PictureList[ViewPt, #]&,
lines1[t]], 1]];

PerspectiveDrawing[ViewPt_] :=
ParametricPlot[Evaluate[
Objectlines[ViewPt, cat]],
{cat, 0.1, 0.918134}, Axes->True, PlotStyle->RGBColor[0, 1, 0],
PlotRange->{{-11, 11}, {-20, 10}},
AspectRatio->Automatic];

```

Appendix D

Program for creating mirror anamorphograms

■ **Note:** The basic mirror programming here was done by Simon Wotherspoon when I first began this project.

■ Cylinder

```
MirrorSurface[xpt_List]:=xpt[[1]]^2+xpt[[2]]^2-9;
```

■ Cone

```
MirrorSurface[xpt_List]:=xpt[[1]]^2+xpt[[2]]^2-1/3*xpt[[3]]^2;
```

```
MirrorSurface[xpt_List]:=xpt[[1]]^2+xpt[[2]]^2-1/3*(xpt[[3]]-10)^2;
```

■ Plane

```
MirrorSurface[xpt_List]:=xpt[[2]]
```

■ Sphere

```
MirrorSurface[xpt_List]:=xpt[[1]]^2+xpt[[2]]^2+(xpt[[3]]-6)^2-36;
```

```
ViewPt:={-2.875,-30,30};
```

```
ObjectPlane[xpt_List]:=xpt[[3]];
```

```
ViewLine[xpt_List,t_]:=ViewPt+t*(xpt-ViewPt);
```

```
Grad[f_] := {D[f,x],D[f,y],D[f,z]};
```

```
SurfaceNormal[xpt_List]:=Grad[MirrorSurface[{x,y,z}]] /.  
{x->xpt[[1]],y->xpt[[2]],z->xpt[[3]]};
```

```
SurfaceIntersectionList[xpt_List]:=zt /.  
Solve[MirrorSurface[ViewLine[xpt,zt]]==0,zt];
```

```
IntersectionList[xpt_List]:=Map[ViewLine[xpt,#]&,  
List[Min[SurfaceIntersectionList[xpt]]]]/N;
```

```
CalcRefLine[xpt_List,IntersectPt_List,t_]:=Block[{normal,u,v},  
n=SurfaceNormal[IntersectPt];  
u=(xpt-ViewPt);  
v=u - 2 * (u . n)/(n . n)*n;  
IntersectPt + t * v  
];
```

```
AllRefLines[xpt_List,t_]:=Map[CalcRefLine[xpt,#,t]&,  
IntersectionList[xpt]]
```



```
CalcObjectPt[refline_List,t_]:=Flatten[refline /.  
Solve[ObjectPlane[refline]==0,t  
ObjectPtList[xpt_List]:=Map[CalcObjectPt[#,t]&,AllRefLines[xpt,t]];
```

Appendix E

Program for creating Single-Image Stereogram of an ellipsoid

□ This program relies on the equation of the circle of intersection of the plane through the viewer's eyes and the row of the Single-Image Stereogram at a height $z=k$ being known. That is, in the case of the sphere with equation $x^2 + (y-a)^2 + z^2 = r^2$ the circle on the plane of intersection is given by $x^2 + (y' - ad/(d^2 + k^2)^{1/2})^2 = r^2 - a^2 + (a^2 d^2 / (d^2 + k^2))$. That is, we have effectively rotated the axes about the original x-axis to obtain a new y'-axis. "d" represents the distance of the viewer from the stereogram. "a" represents the y-co-ordinate of the centre of the sphere.

□ In the case of the ellipsoid with equation $x^2/b^2 + (y-a)^2/c^2 + z^2/f^2 = 1$ the figure on the plane of intersection is given by $x^2 / (b^2 * ((f^2 d^2 + c^2 k^2 - a^2 k^2) / (f^2 d^2 + c^2 k^2))) + (y' - (a * d * f^2 * \text{Sqrt}[d^2 + k^2] / (f^2 d^2 + c^2 k^2)))^2 / ((c^2 f^2 (d^2 + k^2) / (f^2 d^2 + c^2 k^2))) * ((f^2 d^2 + c^2 k^2 - a^2 k^2) / (f^2 d^2 + c^2 k^2)) = 1$; We have effectively rotated the axes about the original x-axis to obtain a new y'-axis.

The following explanations are written for the special case of the sphere. It is relatively straight forward to amend this program for the more general ellipsoid, given the equations above. As mentioned in the text, we must choose a,b,c and f carefully so that the ellipsoid does not lie too close to the plane of the stereogram.

jr.....gives two intersection points with the circle of any line with slope m through the right eye, at (w,0,0). w represents half the eye-spacing of the viewer.

```
jr[a_,r_,d_,w_,k_] :=
{x,y}/.Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2==
r^2-a^2+(a^2*d^2/(d^2+k^2)),y==m*x-m*w},{x,y}]
```

jl.....gives two intersection points with the circle of any line with slope m through the left eye, at (-w,0,0). w represents half the eye-spacing of the viewer.

```
jl[a_,r_,d_,w_,k_] :=
{x,y}/.Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2==
r^2-a^2+(a^2*d^2/(d^2+k^2)),y==m*x+m*w},{x,y}]
```

sloper.....gives the two possible slope values which make the viewlines from the right eye, tangents to the circle.

```
sloper[a_,r_,d_,w_,k_] :=
m/.Solve[jr[a,r,d,w,k][[1,1]]==jr[a,r,d,w,k][[2,1]],m]
```

slopel.....gives the two possible slope values which make the viewlines from the left eye, tangents to the circle.

```

sloper[a_,r_,d_,w_,k_]:=
m/.Solve[j1[a,r,d,w,k][[1,1]]==j1[a,r,d,w,k][[2,1]],m]

```

fn1r.....gives the x-co-ordinate of the intersection point of a right eye tangent line with the SIS which lies on the line $y'=(d^2+k^2)^{1/2}$. d is the distance of the viewer from the SIS and k is the z co-ordinate of the appropriate row of the SIS in the original co-ordinate system.

```

fn1r[a_,r_,d_,w_,k_]:=
x/.
Solve[sloper[a,r,d,w,k][[1]]*x-sloper[a,r,d,w,k][[1]]*w==
Sqrt[d^2+k^2],x]

```

fn2r.....gives the x-co-ordinate of the intersection point of a second right eye tangent line with the SIS which lies on the line $y'=(d^2+k^2)^{1/2}$. Again d is the distance of the viewer from the SIS and k is the z co-ordinate of the appropriate row of the SIS in the original co-ordinate system.

```

fn2r[a_,r_,d_,w_,k_]:=
x/.Solve[sloper[a,r,d,w,k][[2]]*x-sloper[a,r,d,w,k][[2]]*w==
Sqrt[d^2+k^2],x]

```

fn1l.....gives the x-co-ordinate of the intersection point of one left eye tangent line with the SIS which lies on the line $y'=(d^2+k^2)^{1/2}$. Again d is the distance of the viewer from the SIS and k is the z co-ordinate of the appropriate row of the SIS in the original co-ordinate system.

```

fn1l[a_,r_,d_,w_,k_]:=
x/.Solve[sloper[a,r,d,w,k][[1]]*x+sloper[a,r,d,w,k][[1]]*w==
Sqrt[d^2+k^2],x]

```

fn2l.....gives the x-co-ordinate of the intersection point of a second left eye tangent line with the SIS which lies on the line $y'=(d^2+k^2)^{1/2}$. Again d is the distance of the viewer from the SIS and k is the z co-ordinate of the appropriate row of the SIS in the original co-ordinate system.

```

fn2l[a_,r_,d_,w_,k_]:=x/.Solve[sloper[a,r,d,w,k][[2]]*x+
sloper[a,r,d,w,k][[2]]*w==Sqrt[d^2+k^2],x]

```

intang.....gives the intersection point of the two tangents; one from the right eye to the left side of the circle and the other from the left eye to the right side of the circle. This is found by looking at the intersection points of the 4 possible tangents with the SIS. This gives the slope of the appropriate tangent lines.

```

intang[a_,r_,d_,w_,k_] := Module[{slr,sll}, If[N[fn1r[
a,r,d,w,k][[1]] < fn2r[a,r,d,w,k][[1]]], slr=sloper
[a,r,d,w,k][[1]], slr=sloper[a,r,d,w,k][[2]];
If[N[fn1l[a,r,d,w,k][[1]] > fn2l[a,r,d,w,k][[1]]],
sll=sloperl[a,r,d,w,k][[1]], sll=sloperl[a,r,d,w,k][[2]];
{x,y}/.Solve[{y==slr*(x-w), y==sll*(x+w)}, {x,y}]/N

```

tangptlr..... gives the left hand side tangent point of the
viewline from the right eye to the circle.

```

tangptlr[a_,r_,d_,w_,k_] := Module[{tangentptr},
If[N[fn1r[a,r,d,w,k][[1]] < fn2r[a,r,d,w,k][[1]]],
tangentptr={x,y}/.Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2
==r^2-a^2+(a^2*d^2/(d^2+k^2)),
y==sloper[a,r,d,w,k][[1]]*x-sloper[a,r,d,w,k][[1]]*w},
{x,y}], tangentptr=
{x,y}/.Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2
==r^2-a^2+(a^2*d^2/(d^2+k^2)),
y==sloper[a,r,d,w,k][[2]]*x-sloper[a,r,d,w,k][[2]]*w},
{x,y}]]]/N

```

tangptll..... gives the right hand side tangent point of the
viewline from the left eye to the circle.

```

tangptll[a_,r_,d_,w_,k_] := Module[{tangentptl},
If[N[fn1l[a,r,d,w,k][[1]] > fn2l[a,r,d,w,k][[1]]], tangentptl={x,y}/.
Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2==
r^2-a^2+(a^2*d^2/(d^2+k^2)),
y==sloperl[a,r,d,w,k][[1]]*x+sloperl[a,r,d,w,k][[1]]*w},
{x,y}], tangentptl={x,y}/.Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2==
r^2-a^2+(a^2*d^2/(d^2+k^2)),
y==sloperl[a,r,d,w,k][[2]]*x+sloperl[a,r,d,w,k][[2]]*w},
{x,y}]]]/N

```

firstdotr.....is the intersection point with the z=k row of
the SIS of the viewline from the left eye to the tangent
point on the left of the sphere. This tangent point is that
of the viewline from the right eye.

```

firstdotr[a_,r_,d_,w_,k_] :=
x/.Solve[
tangptlr[a,r,d,w,k][[1,2]](x+w)/(tangptlr[a,r,d,w,k][[1,1]]+w)
==Sqrt[d^2+k^2], x]/N

```

lastdotl.....is the intersection point with the z=k row of
the SIS of the viewline from the left eye to its tangent
point on the right of the sphere. This is the last dot used
by the left eye to view the sphere as it moves to the right.
Any dot further to the right can be used for surroundings
by the left eye.

```

lastdot1[a_,r_,d_,w_,k_] :=
x/.Solve[
tangpt11[a,r,d,w,k][[1,2]](x+w)/(tangpt11[a,r,d,w,k][[1,1]]+w)
==Sqrt[d^2+k^2],x]/N

```

lastdot.....is the intersection point with the $z=k$ row of the SIS of the viewline from the right eye to the tangent point on the right of the sphere. This tangent point is that of the viewline from the left eye.

```

lastdot[a_,r_,d_,w_,k_] :=
x/.Solve[
tangpt11[a,r,d,w,k][[1,2]](x-w)/(tangpt11[a,r,d,w,k][[1,1]]-w)
==Sqrt[d^2+k^2],x]/N

```

mainfun1.....This maps the x-co-ord of a given dot,Intr, for the right eye onto the x-co-ord of the dot on the SIS (of our circle in this case) which is the corresponding dot for the right eye when the left eye views Intr.

Note: Since f is a circle in this case there will be two intersection points of the viewline from $l=(-w,0)$. If we want the SIS of the front of the sphere then we choose the intersection point with f which has the smaller y' co-ordinate.(vice-versa for the inside of the sphere)

```

mainfun1[a_,r_,d_,w_,k_][Intr_] :=
(pt={x,y}/.Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2==
r^2-a^2+(a^2*d^2/(d^2+k^2)),y==
(Sqrt[k^2+d^2]/(Intr+w))*x+(Sqrt[k^2+d^2]/(Intr+w))*w}
,{x,y}];
If[pt[[1,2]]<pt[[2,2]],curvept=pt[[1]],curvept=pt[[2]];
x/.Solve[(curvept[[2]]/(curvept[[1]]-w))(x-w)-
(k^2+d^2)^(1/2)==0,x]/N)

```

mainfun2.....This maps the x-co-ord of a given dot,Intl, for the left eye onto the x-co-ord of the dot on the SIS (of our circle in this case) which is the corresponding dot for the left eye when the right eye views Intl.

Note: Since f is a circle in this case there will be two intersection points of the viewline from $r=(w,0)$. If we want the SIS of the front of the sphere then we choose the intersection point with f which has the smaller y' co-ordinate.(vice-versa for the inside of the sphere)

```

mainfun2[a_,r_,d_,w_,k_][Intl_] := (pt =
{x,y}/.Solve[{x^2+(y-(a*d/Sqrt[d^2+k^2]))^2==
r^2-a^2+(a^2*d^2/(d^2+k^2)),y==(Sqrt[k^2+d^2]/
(Intl-w))*x-(Sqrt[k^2+d^2]/
(Intl-w))*w},{x,y}];
If[pt[[1,2]]<pt[[2,2]],curvept=pt[[1]],curvept=pt[[2]];
x/.Solve[(curvept[[2]]/(curvept[[1]]+w))(x+w)-
(k^2+d^2)^(1/2)==0,x]/N)

```

mainfun3.....This maps the x-co-ord of a given dot,Intl, for the left eye onto the x-co-ord of the dot on the SIS (of a background plane in this case..here the plane is $y=38$) which is the corresponding dot for the left eye when the right eye views Intl.

```

mainfun3[a_,r_,d_,w_,k_][Intl_] :=
(pt={x,y}/.Solve[{y==Sqrt[38^2+(k+(38-d)ArcTan[k/d])^2]
,y==(Sqrt[k^2+d^2]/(Intl-w))*x-(Sqrt[k^2+d^2]/
(Intl-w))*w},{x,y}];
x/.Solve[(pt[[1,2]]/(pt[[1,1]]+w))(x+w)-(k^2+d^2)^(1/2)==0,x]/N)

```

mainfun4.....This maps the x-co-ord of a given dot,Intr, for the right eye onto the x-co-ord of the dot on the SIS (of a background plane in this case... $y=38$) which is the corresponding dot for the right eye when the left eye views Intr.

```

mainfun4[a_,r_,d_,w_,k_][Intr_] :=
(pt={x,y}/.Solve[{y==Sqrt[38^2+(k+(38-d)ArcTan[k/d])^2]
,y==
(Sqrt[k^2+d^2]/(Intr+w))*x+(Sqrt[k^2+d^2]/(Intr+w))*w
,{x,y}];
x/.Solve[(pt[[1,2]]/(pt[[1,1]]-w))(x-w)-
(k^2+d^2)^(1/2)==0,x]/N)

```

dotsr.....maps the x co-ord of a given dot from left to right across the row of the SIS by repeatedly applying **mainfun1** until it exceeds **lastdotl**. We can go no further to the right since the viewline from the left eye misses the sphere for any dot farther right. (Nasty messages would appear!).

```

dotsr[a_,r_,d_,w_,k_][x_] := Module[{q=x,elephant={x},
giraffe=lastdotl[a,r,d,w,k][[1]]},
While[(q=mainfun1[a,r,d,w,k][q][[1]])<=giraffe,
AppendTo[elephant,q]];elephant ]

```

dotsl.....maps the x co-ord of a given dot from right to left across the row of the SIS by repeatedly applying **mainfun2** until it is smaller than **mainfun1**[**firstdotr**]. This **mainfun1**[**firstdotr**] when we move from right to left corresponds to **lastdotl** when we move from left to right.

```

dotsl[a_,r_,d_,w_,k_][x_]:=Module[{q=x,elephant={x},
rabbit=mainfun1[a,r,d,w,k][firstdotr[a,r,d,w,k][[1]]][[1]]},
While[(q=mainfun2[a,r,d,w,k][q][[1]])>=rabbit,
AppendTo[elephant,q]];elephant ]

```

rhino.....takes the last dot of dotsr[x] and maps it one step further to the right than lastdot. This cannot be done with the function dotsr because in applying the test "Is it bigger than lastdot?" etc. an attempt would be made to have the viewline from the left eye intersect the circle which is not possible for dots with x-co-ord bigger than lastdotl.

```

rhino[a_,r_,d_,w_,k_][x_]:=
mainfun1[a,r,d,w,k][Take[dotsr[a,r,d,w,k][x],-1][[1]]]/N

```

rhinol.....takes the last dot of dotsl[x] and maps it one step further to the left than mainfun1[firstdotr]. This cannot be done with the function dotsl because in applying the test "Is it smaller than mainfun1[firstdotr]?" etc. an attempt would be made to have the viewline from the right eye intersect the circle which is not possible for dots with x-co-ord smaller than mainfun1[firstdotr].

```

rhinol[a_,r_,d_,w_,k_][x_]:=
mainfun2[a,r,d,w,k][Take[dotsl[a,r,d,w,k][x],-1][[1]]]/N

```

hippo.....takes rhino[x] which is in the last possible interval to the right used to see the circle (only by the right eye) and keeps mapping this dot to the right to give the x co-ords. of dots which enable the viewer to see a surrounding or background plane up to the specified value of x on the SIS. i.e. rhino[x] will be used by the left eye to see the plane rather than the circle.

```

hippo[a_,r_,d_,w_,k_,maxx_][x_]:=
Module[{q=Take[rhino[a,r,d,w,k][x],-1][[1]],croc={}},
While[(q=mainfun4[a,r,d,w,k][q][[1]])<=maxx,
AppendTo[croc,q]];croc ]/N

```

hippol.....takes rhino[x] which is in the last possible interval to the left used to see the circle (only by the left eye) and keeps mapping this dot to the left to give the x co-ords. of dots which enable the viewer to see a surrounding or background plane up to the specified value of x on the SIS. i.e. rhino[x] will be used by the right eye to see the line (of plane) rather than the circle.

```

hippol[a_,r_,d_,w_,k_,minx_][x_]:=
Module[{q=Take[rhinol[a,r,d,w,k][x],-1][[1]],croc={}},
While[(q=mainfun3[a,r,d,w,k][q][[1]])>=minx,
AppendTo[croc,q]];croc ]/N

```


rowdots.....takes the union of dots from left to right to give the x co-ords. of a row of dots which allow the viewer to perceive from the left of the circle to the line on the right which is on the background plane.

```
rowdots[a_,r_,d_,w_,k_,maxx_][x_] :=
Join[dotsr[a,r,d,w,k][x],rhino[a,r,d,w,k][x],
hippo[a,r,d,w,k,maxx][x]]
```

rowdotsl.....takes the union of dots from right to left to give the x co-ords. of a row of dots which allow the viewer to perceive from the right of the circle to the line on the left which is on the background plane.

```
rowdotsl[a_,r_,d_,w_,k_,minx_][x_] :=
Join[dotsl[a,r,d,w,k][x],rhinol[a,r,d,w,k][x],
hippol[a,r,d,w,k,minx][x]]
```

edge.....is an iterative function which maps the x-co-ord. of a dot to the left to give a set of dots which enable the viewer to see a surrounding plane on the left of the sphere.

```
edge[a_,r_,d_,w_,k_,minx_][x_] := Module[{b=x,q={}},
While[(b=mainfun3[a,r,d,w,k][b][[1]]) >= minx,
AppendTo[q,b]];q]
```

edgel.....is an iterative function which maps the x-co-ord. of a dot to the right to give a set of dots which enable the viewer to see a surrounding plane on the right of the sphere.

```
edgel[a_,r_,d_,w_,k_,maxx_][x_] := Module[{b=x,q={}},
While[(b=mainfun4[a,r,d,w,k][b][[1]]) <= maxx,
AppendTo[q,b]];q]
```

intvalr.....gives 3 (arbitrary...depends on scale and dot-size) random real numbers between the x co-ord. of one dot and the x co-ord. of its matching dot to the right for the sphere.

```
intvalr[a_,r_,d_,w_,k_,xmin_] := Table[Random[Real,
{xmin,mainfun1[a,r,d,w,k][xmin][[1]]}],
{3}]/N
```

intvall.....gives 3 (arbitrary...depends on scale and dot-size) random real numbers between the x co-ord. of one dot and the x co-ord. of its matching dot to the left for the sphere.

```
intvall[a_,r_,d_,w_,k_,xmax_] := Table[Random[Real,
{xmax,mainfun2[a,r,d,w,k][xmax][[1]]}],
{3}]/N
```

commencer....gives an interval of x-co-ordinates of dots from which our SIS can be generated. It chooses three (this is arbitrary...depends on scale and dot-size) pseudorandom

real numbers between the x-co-ordinate of one dot and the x-co-ordinate of its matching dot to the right for the sphere. The random number generator is seeded with the integer n so that it is possible to obtain the same sequence of numbers in a repeat run of the program. The given dot is included in the initial interval here. There are cases when this should be eliminated.

```
commencer[a_,r_,xmin_,n_][d_,w_,k_] :=
(SeedRandom[n+Ceiling[5*k]];intvalr[a,r,d,w,k,xmin]
//N;
Join[{xmin},intvalr[a,r,d,w,k,xmin]])
```

commencel....gives an interval of x-co-ordinates of dots from which our SIS can be generated. It chooses three (this is arbitrary...depends on scale and dot-size) pseudorandom real numbers between the x-co-ordinate of one dot and the x-co-ordinate of its matching dot to the left for the sphere. The random number generator is seeded with the integer n so that it is possible to obtain the same sequence of numbers in a repeat run of the program. The given dot is included in the initial interval here. There are cases when this should be eliminated.

```
commencel[a_,r_,xmax_,n_][d_,w_,k_] :=
(SeedRandom[n+Ceiling[5*k]];intvall[a,r,d,w,k,xmax]
//N;
Join[{xmax},intvall[a,r,d,w,k,xmax]])
```

ptsr.....maps both the functions rowdots and edge onto the initial interval,commencer,to give a set of x-co-ords of a row of dots for the sphere. The appropriate z-co-ordinate is appended to give a set of points in the x-z plane.

```
ptsr[a_,r_,xmin_,n_,d_,w_,minx_,maxx_][k_] :=
(spots=Join[Flatten[
Map[rowdots[a,r,d,w,k,maxx],commencer[a,r,xmin,n][d,w,k]],1],
Flatten[Map[edge[a,r,d,w,k,minx],
commencer[a,r,xmin,n][d,w,k]],1]];
Map[Append[#,k]&,Partition[spots,1]]);
```

ptsl.....maps both the functions rowdotsl and edgel onto the initial interval,commencel,to give a set of x-co-ords of a row of dots for the sphere. The appropriate z-co-ordinate is appended to give a set of points in the x-z plane.

```

ptsl[a_,r_,xmax_,n_,d_,w_,minx_,maxx_][k_]:=
(splots=Join[Flatten[
Map[rowdotsl[a,r,d,w,k,minx],
commencel[a,r,xmax,n][d,w,k]],1],
Flatten[Map[edgel[a,r,d,w,k,maxx],
commencel[a,r,xmax,n][d,w,k]],1]]];
Map[Append[#,k]&,Partition[splots,1]]];

```

zmin.....gives the minimum z-co-ordinate for which the circle of intersection of the sloping plane with the sphere has radius 0. i.e.the z-value of the bottom tangent point of this plane.

```

zmin[a_,d_,r_]:=
Min[q/.Solve[r^2-a^2+(a^2*d^2/(d^2+q^2))==0,q]]/N

```

zmax.....gives the maximum z-co-ordinate for which the circle of intersection of the sloping plane with the sphere has radius 0. i.e.the z-value of the top tangent point of this plane.

```

zmax[a_,d_,r_]:=
Max[q/.Solve[r^2-a^2+(a^2*d^2/(d^2+q^2))==0,q]]/N

```

plintvalr.....gives 3 (arbitrary...depends on scale and dot-size) random real numbers between the x co-ord. of one dot and the x co-ord. of its matching dot to the right for the surrounding plane.

```

plintvalr[a_,r_,d_,w_,k_,xmin_]:=Table[Random[Real,
{xmin,mainfun4[a,r,d,w,k][xmin][1]}],
{3}]/N

```

plintvall.....gives 3 (arbitrary...depends on scale and dot-size) random real numbers between the x co-ord. of one dot and the x co-ord. of its matching dot to the left for the surrounding plane.

```

plintvall[a_,r_,d_,w_,k_,xmax_]:=Table[Random[Real,
{xmax,mainfun3[a,r,d,w,k][xmax][1]}],
{3}]/N

```

plcommencer....gives an interval of x-co-ordinates of dots from which our SIS can be generated. It chooses three (this is arbitrary...depends on scale and dot-size) pseudorandom real numbers between the x-co-ordinate of one dot and the x-co-ordinate of its matching dot to the right for the surrounding plane.

The random number generator is seeded with the integer n so that it is possible to obtain the same sequence of numbers in a repeat run of the program. The given dot is included in the initial interval here.

```
plcommencer[a_,r_,xmin_,n_][d_,w_,k_] :=
(SeedRandom[n+Ceiling[5*k]];plintvalr[a,r,d,w,k,xmin]
//N;
Join[{xmin},plintvalr[a,r,d,w,k,xmin]])
```

plcommencer....gives an interval of x-co-ordinates of dots from which our SIS can be generated. It chooses three (this is arbitrary...depends on scale and dot-size) pseudorandom real numbers between the x-co-ordinate of one dot and the x-co-ordinate of its matching dot to the left for the surrounding plane.

The random number generator is seeded with the integer n so that it is possible to obtain the same sequence of numbers in a repeat run of the program. The given dot is included in the initial interval here.

```
plcommencerl[a_,r_,xmax_,n_][d_,w_,k_] :=
(SeedRandom[n+Ceiling[5*k]];plintvall[a,r,d,w,k,xmax]
//N;
Join[{xmax},plintvall[a,r,d,w,k,xmax]])
```

pldotsr.....maps the x co-ord of a given dot from left to right across the row of the SIS by repeatedly applying mainfun4 (for the plane on the right) until it exceeds a given maximum x-value.

```
pldotsr[a_,r_,d_,w_,k_,maxx_][x_] := Module[{q=x,elephant={x}},
While[(q=mainfun4[a,r,d,w,k][q][[1]])<=maxx,
AppendTo[elephant,q]];elephant ]
```

pldotsl.....maps the x co-ord of a given dot from right to left across the row of the SIS by repeatedly applying mainfun3 (for the plane on the left) until it is smaller than a given minimum x-value.

```
pldotsl[a_,r_,d_,w_,k_,minx_][x_] := Module[{q=x,elephant={x}},
While[(q=mainfun3[a,r,d,w,k][q][[1]])>=minx,
AppendTo[elephant,q]];elephant ]
```

plptsr.....maps the function pldotsr onto the initial interval, plcommencer, to give a set of x-co-ords of a row of dots for the surrounding plane on the right. The appropriate z-co-ordinate is appended to give a set of points in the x-z plane.

```
plptsr[a_,r_,xmin_,n_,d_,w_,maxx_][k_] := (spots=Flatten[
Map[pldotsr[a,r,d,w,k,maxx],plcommencer[a,r,xmin,n][d,w,k]],1];
Map[Append[#,k]&,Partition[spots,1]]);
```

plptsl.....maps the function pldotsl onto the initial interval, plcommencerl, to give a set of x-co-ords of a row of dots for the

surrounding plane on the left. The appropriate z-co-ordinate is appended to give a set of points in the x-z plane.

```
plpts1[a_,r_,xmax_,n_,d_,w_,minx_][k_]:=({plots=Flatten[
Map[pldots1[a,r,d,w,k,minx],plcommence1[a,r,xmax,n][d,w,k]],1];
Map[Append[#,k]&,Partition[plots,1]]);
```

stereog..... calculates the set of points for each row of the SIS and then plots them with the options specified.

Consideration must be given to the dot size, the difference in the z-co-ordinate of each row, the colours required.e.g.Do we need or want a different colour for each row? Why?

What size do we want the SIS to be?

Must it fit on an A4 page? etc.Can we change the shape of the dots ?

Since the program is calculating rows from both the right and the left, the minimum z-values are adjusted for each section of the SIS so that the rows do not completely overlap. The advantage of calculating in both directions is our ability to start each iteration with a dot that is definitely on the visible boundary of the sphere. The last dot in each row is close to, but not on, the boundary.

□ **Note:**The step size must be such that the z-co-ordinate is never 0 since z sometimes appears on the denominator

```
stereog[a_,r_,minx_,maxx_,n_,d_,w_,steps_]:=
ListPlot[Join[Flatten[Table[plptsr[a,r,minx,n,d,w,
maxx][k],
{k,minx,zmin[a,d,r]-0.02,steps}],1],
Flatten[Table[plptsr[a,r,minx,n,d,w,maxx][k],
{k,zmax[a,d,r]+0.02,maxx,steps}],1],
Flatten[Table[ptsr[a,r,firstdotr[a,r,d,w,k][[1]],n,d,w,
mina,maxx][k],
{k,zmin[a,d,r]+0.1,zmax[a,d,r]-0.1,steps}],1],
Flatten[Table[plpts1[a,r,maxx,n,d,w,minx][k],
{k,minx+0.05,zmin[a,d,r]-0.02,steps}],1],
Flatten[Table[plpts1[a,r,maxx,n,d,w,minx][k],
{k,zmax[a,d,r]+0.05,maxx,steps}],1],
Flatten[Table[pts1[a,r,lastdot[a,r,d,w,k][[1]],n,d,w,
minx,maxx][k],
{k,zmin[a,d,r]+0.07,zmax[a,d,r]-0.07,steps}],1],
Prolog->AbsolutePointSize[2],
PlotStyle->RGBColor[0,0,0.9],
Axes->False,PlotRange->{{minx,maxx},{minx,maxx}},
AspectRatio->Automatic]
```

- **Note:**The edit style must be appropriate to the size of the SIS so that it is exactly to scale (in cm).This is because the program is written using the set eye-spacing of the viewer (approx. 5.75cm).Any shrinkage or enlargement causes distortion of the perceived picture.

stereog[35,3,-4,4,143,30,2875/1000,2/10]

Appendix F

Program for creating Single-Image Stereogram of a cube

ViewPt gives the co-ordinates of the midpoint of the line segment joining the nodal points of the viewer's eyes. In this case it is at the origin of our co-ordinate system. It is used to find the nearest vertices to the viewer.

```
ViewPt:={0,0,0};
```

rh and **uh** define rotation matrices using homogeneous co-ordinates as detailed in Appendix A. In this case **rh** is the composition of a rotation by **ang3**, firstly about the z-axis, and then about the y-axis. **uh** is the composition of a rotation by **ang2** about the x-axis and then by an angle of **ang1** about the y-axis.

```
rh[ang3_] := {{Cos[ang3], Sin[ang3], 0, 0}, {-Sin[ang3], Cos[ang3], 0, 0},  
{0, 0, 1, 0}, {0, 0, 0, 1}}.  
{{Cos[ang3], 0, Sin[ang3], 0}, {0, 1, 0, 0}, {-Sin[ang3], 0, Cos[ang3], 0}  
, {0, 0, 0, 1}}  
  
uh[ang2_, ang1_] := {{1, 0, 0, 0}, {0, Cos[ang2], Sin[ang2], 0},  
{0, -Sin[ang2], Cos[ang2], 0}, {0, 0, 0, 1}}.  
{{Cos[ang1], 0, Sin[ang1], 0}, {0, 1, 0, 0}, {-Sin[ang1], 0, Cos[ang1], 0},  
{0, 0, 0, 1}}
```

scale is a scaling matrix for scaling by a factor of **Si** in the i-direction.

```
scale[Sx_, Sy_, Sz_] := {{Sx, 0, 0, 0}, {0, Sy, 0, 0}, {0, 0, Sz, 0}, {0, 0, 0, 1}}
```

transl is a translation matrix for translating by **Tj** units in the j-direction.

```
transl[Tx_, Ty_, Tz_] := {{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {Tx, Ty, Tz, 1}}
```

netransf is the net transformation matrix.

```
netransf[Sx_, Sy_, Sz_, Tx_, Ty_, Tz_, ang1_, ang2_, ang3_] :=  
Map[Drop[#, -1] &, rh[ang3].uh[ang2, ang1].scale[Sx, Sy, Sz].  
transl[Tx, Ty, Tz]]
```

corners2 and **corners** give the co-ordinates of the vertices, once the cube has been rotated and translated in the required manner.

The order of these vertices is important.

```
corners2[Sx_, Sy_, Sz_, Tx_, Ty_, Tz_, ang1_, ang2_, ang3_, dim_] :=  
Flatten[Table[{{-dim, j, -dim, 1},  
netransf[Sx, Sy, Sz, Tx, Ty, Tz, ang1, ang2, ang3],  
{-dim, j, dim, 1}.netransf[Sx, Sy, Sz, Tx, Ty, Tz, ang1, ang2, ang3],  
{dim, j, dim, 1}.netransf[Sx, Sy, Sz, Tx, Ty, Tz, ang1, ang2, ang3],  
{dim, j, -dim, 1}.netransf[Sx, Sy, Sz, Tx, Ty, Tz, ang1, ang2, ang3]}],  
{j, -dim, dim, 2*dim}]/N, 1];  
  
corners=corners2[1, 1, 1, 0, 40, 0, Pi/12, Pi/6, Pi/4, 3.5]
```

corns pairs the co-ordinates of each vertex with the co-ordinates of the viewpoint.

```
corns=Map[Prepend[#, ViewPt] &,  
Partition[corners, 1]]
```


distance gives the distance between two points with given co-ordinates.

```
distance[List[v_,x_]]:=Sqrt[(v[[3]]-x[[3]])^2+(v[[2]]-x[[2]])^2+
(v[[1]]-x[[1]])^2]
```

dists gives the list of distances of each vertex from the viewpoint and again a definite order is maintained.

```
dists=Map[distance,corns]
```

closestvert gives the number of the closest vertex. In some examples there maybe more than one closest vertex. **closestvert** chooses the first in the list of possibilities.

```
closestvert=Flatten[Position[dists,Min[dists]],1][[1]]
```

If $x_$ denotes the vertex number according to our specific ordering then **adjacentvertices** gives the numbers of the vertices which have an edge in common with x .

```
adjacentvertices[x_]:=If[x==1,{2,4,5},If[x==2,{1,3,6},
If[x==3,{2,4,7},If[x==4,{1,3,8},If[x==5,{1,6,8},If[x==6,
{2,5,7},If[x==7,{3,6,8},{4,5,7}]]]]]]]
```

dotproduct gives the dot product of two vectors and **dps** gives the list of dotproducts of the vectors from the viewpoint to the closest vertex and from the closest vertex to its adjacent vertices.

```
dotproduct[x_]:=
(corners[x]-corners[closestvert]).
(ViewPt-corners[closestvert])
```

```
dps=Map[dotproduct,adjacentvertices[closestvert]]
```

The number of visible faces is given by the number of dot products less than 0 (given by **neg**). This uses one of our results see.....

```
neg=Select[Map[dotproduct,adjacentvertices[closestvert]],#<0&]
```

position gives the position in the list of adjacent vertices, of the vertices which give a negative dotproduct. e.g. position 1 does not necessarily mean vertex 1.

```
position=Flatten[Union[Map[Position[dps,#]&,neg]],2]
```

nvisiblefaces gives the number of visible faces by giving the number of negative dot products.

```
nvisiblefaces=
Length[Select[
Map[dotproduct,adjacentvertices[closestvert]],#<0&]]
```

visiblefaces gives the lists of the vertices co-ordinates for each of the visible faces. There are many cases to consider however, we have shown that the maximum possible number of visible faces is three (3).

```
visiblefaces=If[closestvert==2&&nvisiblefaces==3,
{{corners[[2]],corners[[3]],corners[[7]],corners[[6]]},
{corners[[1]],corners[[2]],corners[[6]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==2&&nvisiblefaces==2&&position=={1,2},
```

```

{{corners[[2]],corners[[3]],corners[[7]],corners[[6]]},
{corners[[2]],corners[[6]],corners[[5]],corners[[1]]}},
If[closestvert==2&&nvisiblefaces==2&&position=={1,3},
{{corners[[2]],corners[[3]],corners[[7]],corners[[6]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==2&&nvisiblefaces==2&&position=={2,3},
{{corners[[1]],corners[[2]],corners[[6]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==2&&nvisiblefaces==1&&position=={1},
{{corners[[2]],corners[[3]],corners[[7]],corners[[6]]}},
If[closestvert==2&&nvisiblefaces==1&&position=={2},
{{corners[[1]],corners[[2]],corners[[6]],corners[[5]]}},
If[closestvert==2&&nvisiblefaces==1&&position=={3},
{{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==1&&nvisiblefaces==3,
{{corners[[1]],corners[[4]],corners[[8]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[6]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==1&&nvisiblefaces==2&&position=={1,2},
{{corners[[1]],corners[[4]],corners[[8]],corners[[5]]},
{corners[[2]],corners[[6]],corners[[5]],corners[[1]]}},
If[closestvert==1&&nvisiblefaces==2&&position=={1,3},
{{corners[[1]],corners[[4]],corners[[8]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==1&&nvisiblefaces==2&&position=={2,3},
{{corners[[1]],corners[[2]],corners[[6]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==1&&nvisiblefaces==1&&position=={1},
{{corners[[1]],corners[[4]],corners[[8]],corners[[5]]}},
If[closestvert==1&&nvisiblefaces==1&&position=={2},
{{corners[[1]],corners[[2]],corners[[6]],corners[[5]]}},
If[closestvert==1&&nvisiblefaces==1&&position=={3},
{{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==3&&nvisiblefaces==3,
{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]},
{corners[[2]],corners[[3]],corners[[7]],corners[[6]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==3&&nvisiblefaces==2&&position=={1,2},
{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]},
{corners[[2]],corners[[3]],corners[[7]],corners[[6]]}},
If[closestvert==3&&nvisiblefaces==2&&position=={1,3},
{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==3&&nvisiblefaces==2&&position=={2,3},
{{corners[[2]],corners[[3]],corners[[7]],corners[[6]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==3&&nvisiblefaces==1&&position=={1},
{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]}},
If[closestvert==3&&nvisiblefaces==1&&position=={2},
{{corners[[2]],corners[[3]],corners[[7]],corners[[6]]}},
If[closestvert==3&&nvisiblefaces==1&&position=={3},
{{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==4&&nvisiblefaces==3,

```

```

{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]},
{corners[[1]],corners[[4]],corners[[8]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==4&&nvisiblefaces==2&&position=={1,2},
{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]},
{corners[[1]],corners[[4]],corners[[8]],corners[[5]]}},
If[closestvert==4&&nvisiblefaces==2&&position=={1,3},
{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==4&&nvisiblefaces==2&&position=={2,3},
{{corners[[1]],corners[[4]],corners[[8]],corners[[5]]},
{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==4&&nvisiblefaces==1&&position=={1},
{{corners[[3]],corners[[7]],corners[[8]],corners[[4]]}},
If[closestvert==4&&nvisiblefaces==1&&position=={2},
{{corners[[1]],corners[[4]],corners[[8]],corners[[5]]}},
If[closestvert==4&&nvisiblefaces==1&&position=={3},
{{corners[[1]],corners[[2]],corners[[3]],corners[[4]]}},
If[closestvert==5&&nvisiblefaces==3,
{{corners[[5]],corners[[8]],corners[[7]],corners[[6]]},
{corners[[1]],corners[[2]],corners[[6]],corners[[5]]},
{corners[[1]],corners[[5]],corners[[8]],corners[[4]]}},
If[closestvert==5&&nvisiblefaces==2&&position=={1,2},
{{corners[[5]],corners[[8]],corners[[7]],corners[[6]]},
{corners[[1]],corners[[2]],corners[[6]],corners[[5]]}},
If[closestvert==5&&nvisiblefaces==2&&position=={1,3},
{{corners[[5]],corners[[8]],corners[[7]],corners[[6]]},
{corners[[1]],corners[[5]],corners[[8]],corners[[4]]}},
If[closestvert==5&&nvisiblefaces==2&&position=={2,3},
{{corners[[1]],corners[[2]],corners[[6]],corners[[5]]},
{corners[[1]],corners[[5]],corners[[8]],corners[[4]]}},
If[closestvert==5&&nvisiblefaces==1&&position=={1},
{{corners[[5]],corners[[8]],corners[[7]],corners[[6]]}},
If[closestvert==5&&nvisiblefaces==1&&position=={2},
{{corners[[1]],corners[[2]],corners[[6]],corners[[5]]}},
If[closestvert==5&&nvisiblefaces==1&&position=={3},
{{corners[[1]],corners[[5]],corners[[8]],corners[[4]]}},
If[closestvert==6&&nvisiblefaces==3,
{{corners[[6]],corners[[7]],corners[[8]],corners[[5]]},
{corners[[6]],corners[[7]],corners[[3]],corners[[2]]},
{corners[[2]],corners[[6]],corners[[5]],corners[[1]]}},
If[closestvert==6&&nvisiblefaces==2&&position=={1,2},
{{corners[[6]],corners[[7]],corners[[8]],corners[[5]]},
{corners[[6]],corners[[7]],corners[[3]],corners[[2]]}},
If[closestvert==6&&nvisiblefaces==2&&position=={1,3},
{{corners[[6]],corners[[7]],corners[[8]],corners[[5]]},
{corners[[2]],corners[[6]],corners[[5]],corners[[1]]}},
If[closestvert==6&&nvisiblefaces==2&&position=={2,3},
{{corners[[6]],corners[[7]],corners[[3]],corners[[2]]},
{corners[[2]],corners[[6]],corners[[5]],corners[[1]]}},
If[closestvert==6&&nvisiblefaces==1&&position=={1},
{{corners[[6]],corners[[7]],corners[[8]],corners[[5]]}},
If[closestvert==6&&nvisiblefaces==1&&position=={2},

```

[illegible]

d is the y-co-ordinate of the stereogram i.e.it represents the distance of the viewer from the stereogram in cm.

d=30;

zmin and **zmax** are the minimum and maximum z-co-ordinates on the SIS (top and bottom rows) which enable us to see the face of the cube with vertices **a,b,c,g**. i.e. **a** represents a list of three co-ordinates.

```

zmin[a_,b_,c_,g_] := Min[d*a[[3]]/a[[2]],d*g[[3]]/g[[2]],
d*b[[3]]/b[[2]],d*c[[3]]/c[[2]]];
zmax[a_,b_,c_,g_] := Max[d*a[[3]]/a[[2]],d*g[[3]]/g[[2]],
d*b[[3]]/b[[2]],d*c[[3]]/c[[2]]];

```

bdry gives a parametric representation of the line containing the edge joining vertices **e** and **f**.

intpt gives the three co-ordinates of a point on this same line for any value of the parameter **t**.
parameter gives the value of the parameter, **t**, for which the plane through our eyes and the row of dots at a height **k** intersects the edge between vertices **e** and **f**.

```

bdry[e_,f_,t_] := {x==e[[1]]+(f[[1]]-e[[1]])*t,
                    y==e[[2]]+(f[[2]]-e[[2]])*t,
                    z==e[[3]]+(f[[3]]-e[[3]])*t};
intpt[e_,f_,t_] := {e[[1]]+(f[[1]]-e[[1]])*t,
                    e[[2]]+(f[[2]]-e[[2]])*t,
                    e[[3]]+(f[[3]]-e[[3]])*t};

parameter[k_,e_,f_,t_] := Solve[Join[{-(k y)/d+z==0},
Rest[bdry[e,f,t]]],t,{y,z}];

```

parai assigns to **t** the value obtained for the intersection point in **parameter**. **i** varies from 1 to 4 to distinguish between the different **parameter** values for each edge of the face.

```

para1[k_,a_,b_] := t/.Flatten[parameter[k,a,b,t],1];
para2[k_,b_,c_] := t/.Flatten[parameter[k,b,c,t],1];
para3[k_,c_,g_] := t/.Flatten[parameter[k,c,g,t],1];
para4[k_,g_,a_] := t/.Flatten[parameter[k,g,a,t],1];

```

linei takes the intersection points on the boundaries of the face (usually two for each value of **k**, but for the special case of a vertex could be one only or in the case where the face edge is parallel to the line through the viewer's eyes there could be an infinite number; these extreme cases are unusual and can be eliminated by choosing the step-size so that their exact values of **k** are avoided) and rotates them clock-wise onto the x-y plane. The equation of the line joining them is then found after they have been reduced to points with two co-ordinates.

```

linei[List[a_,b_,c_,g_]][k_]:=
para1[k,a,b];
para2[k,b,c];
para3[k,c,g];
para4[k,g,a];
If[para1[k,a,b]==t,
i1={},If[0<=para1[k,a,b]&&para1[k,a,b]<=1,
i1=intpt[a,b,para1[k,a,b]],i1={}]];
If[para2[k,b,c]==t,
i2={},If[0<=para2[k,b,c]&&para2[k,b,c]<=1,
i2=intpt[b,c,para2[k,b,c]],i2={}]];
If[para3[k,c,g]==t,
i3={},If[0<=para3[k,c,g]&&para3[k,c,g]<=1,
i3=intpt[c,g,para1[k,c,g]],i3={}]];
If[para4[k,g,a]==t,
i4={},If[0<=para4[k,g,a]&&para4[k,g,a]<=1,
i4=intpt[g,a,para4[k,g,a]],i4={}]];
bdryintersections:={i1,i2,i3,i4};
xypts:=Select[bdryintersections,#!={}&];
rotn:={{1,0,0},{0,Cos[ArcTan[N[k/d]]],
-Sin[ArcTan[N[k/d]]]},
{0,Sin[ArcTan[N[k/d]]],Cos[ArcTan[N[k/d]]]}};
newplanepts:=xypts.rotn;
points:=Map[Delete[#,3]&,newplanepts];
If[Min[points[[1,1]],points[[2,1]]]==points[[1,1]],
start:=points[[1]],start:=points[[2]]];
If[start==points[[1]],finish:=points[[2]],
finish:=points[[1]]];
(finish[[2]]-start[[2]])/(finish[[1]]-start[[1]])
x-start[[1]])+start[[2]]

```

st gives the left-hand endpoint of the rotated line and fin gives the right-hand endpoint.

```

st[List[a_,b_,c_,g_]][k_]:= (linei[List[a,b,c,g]][k];start)
fin[List[a_,b_,c_,g_]][k_]:= (linei[List[a,b,c,g]][k];finish)

```

firstdot gives the dot on the RDS which corresponds to st and similarly, lastdot gives the dot that corresponds to fin. Note: the right eye would use firstdot to see either an adjoining face or the background plane and the left eye would use the lastdot similarly.

```

firstdot[d_,w_,k_][List[a_,b_,c_,g_]]:=
x/.Solve[(st[List[a,b,c,g]][k][[2]]/(st[List[a,b,c,g]][k][[1]]+w))
(x+w)-(k^2+d^2)^(1/2)==0,x]

lastdot[d_,w_,k_][List[a_,b_,c_,g_]]:=
x/.Solve[(fin[List[a,b,c,g]][k][[2]]/(fin[List[a,b,c,g]][k][[1]]-w))
(x-w)-(k^2+d^2)^(1/2)==0,x]

```

mainfun takes a point for the right eye and maps it onto the next dot for the right eye as we construct the stereogram for the cube from left to right.

```

mainfun[d_,w_,k_][List[a_,b_,c_,g_]][Intr_]:=
(pt=x/.Solve[((k^2+d^2)^(1/2)/(Intr+w))*(x+w)
-linei[List[a,b,c,g]][k]==0,x];
linept={pt,((k^2+d^2)^(1/2)/(Intr+w))*(pt+w)};
x/.Solve[(linept[[2]]/(linept[[1]]-w))(x-w)-
(k^2+d^2)^(1/2)==0,x]/N);

```

mainfun2 takes a point for the left eye and maps it onto the next dot to the left to enable us to see a background plane, in particular $y=48$ in this case.

```

mainfun2[d_,w_,k_][List[a_,b_,c_,g_]][Intr_]:=
(pt={x,y}/.Solve[{y==Sqrt[41^2+(k+(41-d)ArcTan[k/d])^2],
y==(Sqrt[k^2+d^2]/(Intr-w))*x-(Sqrt[k^2+d^2]/
(Intr-w))*w},{x,y}];
x/.Solve[(pt[[1,2]]/(pt[[1,1]]+w))(x+w)-(k^2+d^2)^(1/2)==0,x]/N)

```

mainfun3 takes a point for the right eye and maps it onto the next dot to the right to enable us to see a background plane, in particular $y=48$ in this case.

```

mainfun3[d_,w_,k_][List[a_,b_,c_,g_]][Intr_]:=
(pt={x,y}/.Solve[{y==Sqrt[41^2+(k+(41-d)ArcTan[k/d])^2],
y==(Sqrt[k^2+d^2]/(Intr+w))*x+(Sqrt[k^2+d^2]/
(Intr+w))*w},{x,y}];
x/.Solve[(pt[[1,2]]/(pt[[1,1]]-w))(x-w)-(k^2+d^2)^(1/2)==0,x]/N)

```

dots maps the x co-ordinate of a given dot from left to right across a row of the RDS by repeatedly applying mainfun until it exceeds the lastdot for the face under consideration at the time.

```

dots[d_,w_,k_][List[a_,b_,c_,g_]][x_]:= Module[{q=x,elephant={x},
rabbit=lastdot[d,w,k][List[a,b,c,g]][[1]]},
While[(q=(mainfun[d,w,k][List[a,b,c,g]][q]))[[1]]<=rabbit,
AppendTo[elephant,q]];elephant ]

```

dots2 maps the x co-ordinate of a given dot from right to left across the row by repeatedly applying mainfun2 until its value is smaller than a given value .(in this case -10)

```

dots2[d_,w_,k_][List[a_,b_,c_,g_]][x_]:= Module[{q=x,croc={x}},
While[(q=(mainfun2[d,w,k][List[a,b,c,g]][q]))[[1]]>=-10,
AppendTo[croc,q]];croc ]

```

dots3 maps the x co-ordinate of a given dot from left to right across the row by repeatedly applying mainfun3 until its value exceeds a given value .(in this case 10)

```

dots3[d_,w_,k_][List[a_,b_,c_,g_]][x_]:= Module[{q=x,giraffe={x}},
While[(q=(mainfun3[d,w,k][List[a,b,c,g]][q]))[[1]]<=10,
AppendTo[giraffe,q]];giraffe ]

```

plintvalr gives 7 (arbitrary... depends on scale and dot-size) random real numbers between the x co-ordinate of one dot and the x-co-ord. of its matching dot to the right for the surrounding plane.

```

plintvalr[d_,w_,k_][List[a_,b_,c_,g_]][xmin_]:=Table[Random[Real,
{xmin,mainfun3[d,w,k][List[a,b,c,g]][xmin]][[1]]],{7}]/N

```

plcommencer gives an interval of x co-ordinates for dots by choosing seven pseudorandom real numbers between the x co-ordinate of one dot and the x-co-ordinate of its matching dot to the

right in this case for the surrounding plane. The random number generator is seeded with the integer n so that it is possible to obtain the same sequence of numbers in a repeat run of the program. The given dot is included in the interval here.

```
plcommencer[d_,w_,k_,n_] [List[a_,b_,c_,g_]] [xmin_] :=
(SeedRandom[n+Ceiling[5*k]]);
plintvalr[d,w,k] [List[a,b,c,g]] [xmin] / N;
Join[{xmin},plintvalr[d,w,k] [List[a,b,c,g]] [xmin]]
```

pldotsr maps the x co-ordinate of a given dot from left to right across the row of the RDS by repeatedly applying **mainfun3** until it exceeds a given maximum.

```
pldotsr[d_,w_,k_] [List[a_,b_,c_,g_]] [x_] := Module[{q=x,flea={x}},
While[(q=(mainfun3[d,w,k] [List[a,b,c,g]] [q]) [[1]]) <= 10,
AppendTo[flea,q]]; flea]
```

plptsr maps the function **pldotsr** onto the initial interval, **plcommencer**, to give a set of x-co-ordinates of a row of dots for the background plane. The appropriate z-co-ord. is appended to give a set of points in the x-z plane.

```
plptsr[d_,w_,k_,n_] [List[a_,b_,c_,g_]] := (splots=Flatten[
Map[pldotsr[d,w,k] [List[a,b,c,g]],
plcommencer[d,w,k,n] [List[a,b,c,g]] [-10]],1];
Map[Append[#,k]&,Partition[splots,1]]);
```

Note: before this we must find the suitable k values using $zmin$ and $zmax$

ju and **ku** give the minimum and maximum z values on the RDS (i.e. values of k) respectively for each visible face of the cube.

```
ju=Flatten[Map[Apply[zmin,#]&,visiblefaces],1]
ku=Flatten[Map[Apply[zmax,#]&,visiblefaces],1]
```

To construct a row of dots for the cube consideration had to be given to the problem of working out for each k , which of the visible faces required dots at that level and in what order. It is possible to have rows of dots which traverse no faces, one face, two faces or three faces even when three faces are visible. **cuberowdots** solves this problem to give a row of x-co-ordinates of dots which start with the far left of the cube and move across the cube to the right and onto the background plane. The background plane to the left is considered separately and is joined later. (Although some thought must be given to amending this to avoid central band with monocular clues.)

To understand how **cuberowdots** works we will consider one case.

Suppose that there are three visible faces and that k lies between the minimum and maximum z -values for two of the visible faces, say 1 and 3. This means that this row of dots only crosses two of the faces. If we start at the left hand side of the cube then we need to find which of these two faces is furthestest to the left. This is done by comparing the **firstdot** for visible faces 1 and 3. **Firstdot** gives an x-co-ord. and so the minimum one is found to tell us which face we consider first. It is then just a matter of starting with **dots[visible face which is far left]** and proceeding until the next face is met (the stopping rule in **dots** ensures that we stop at the correct place). The last dot for the first face is then mapped onto successive dots for the second face using **dots[face which is furthestest right for this row]**. When the edge of this face is

met, the last dot is mapped across to the right to enable us to see the background plane. i.e. **dots3** is employed.

cuberowdots considers all the possible cases which can be explained analogously.

```

cuberowdots[d_,w_,k_,n_][x_] :=
If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]]&&ju[[2]]<=k<=ku[[2]]&&
ju[[3]]<=k<=ku[[3]],
(f1=Flatten[Map[firstdot[d,2.875,k],visiblefaces],1];
vf1=Flatten[Position[f1,Min[f1]],1][[1]];
sd=Drop[f1,{vf1,vf1}];
vf2=Flatten[Position[f1,Min[sd]],1][[1]];
vf3=Flatten[Position[f1,Max[sd]],1][[1]];
Join[f1=dots[d,w,k][visiblefaces[[vf1]]][x],
f2=dots[d,w,k][visiblefaces[[vf2]]][Take[f1,-1][[1]]],
f3=dots[d,w,k][visiblefaces[[vf3]]][Take[f2,-1][[1]]],
dots3[d,w,k][visiblefaces[[vf3]]][Take[f3,-1][[1]]],
Flatten[Map[dots2[d,w,k][visiblefaces[[vf1]]],
commence[d,w,k,n][visiblefaces[[vf1]]],1]]),

If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]]&&ju[[2]]<=k<=ku[[2]],
(f1=Flatten[Map[firstdot[d,2.875,k],
Drop[visiblefaces,{3,3}],1];
vf1=Flatten[Position[f1,Min[f1]],1][[1]];
vf2=Flatten[Position[f1,Max[f1]],1][[1]];
Join[f1=dots[d,w,k][Drop[visiblefaces,{3,3}][[vf1]]][x],
f2=dots[d,w,k][Drop[visiblefaces,{3,3}][[vf2]]][Take[f1,-1][[1]]],
dots3[d,w,k][visiblefaces[[vf2]]][Take[f2,-1][[1]]],
Flatten[Map[dots2[d,w,k][visiblefaces[[vf1]]],
commence[d,w,k,n][visiblefaces[[vf1]]],1]]),

If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]]&&ju[[3]]<=k<=ku[[3]],
(f1=Flatten[Map[firstdot[d,2.875,k],
Drop[visiblefaces,{2,2}],1];
vf1=Flatten[Position[f1,Min[f1]],1][[1]];
vf2=Flatten[Position[f1,Max[f1]],1][[1]];
vf3=Flatten[Position[Flatten[Map[firstdot[d,2.875,k],
visiblefaces],1],Min[f1]],1][[1]];
Join[f1=dots[d,w,k][Drop[visiblefaces,{2,2}][[vf1]]][x],
f2=dots[d,w,k][Drop[visiblefaces,{2,2}][[vf2]]][Take[f1,-1][[1]]],
dots3[d,w,k][visiblefaces[[vf2]]][Take[f2,-1][[1]]],
Flatten[Map[dots2[d,w,k][visiblefaces[[vf3]]],
commence[d,w,k,n][visiblefaces[[vf3]]],1]]),

If[nvisiblefaces==3&&ju[[2]]<=k<=ku[[2]]&&ju[[3]]<=k<=ku[[3]],
(f1=Flatten[Map[firstdot[d,2.875,k],
Drop[visiblefaces,{1,1}],1];
vf1=Flatten[Position[f1,Min[f1]],1][[1]];
vf2=Flatten[Position[f1,Max[f1]],1][[1]];
Join[f1=dots[d,w,k][Drop[visiblefaces,{1,1}][[vf1]]][x],
f2=
dots[d,w,k][Drop[visiblefaces,{1,1}][[vf2]]][Take[f1,-1][[1]]],
dots3[d,w,k][visiblefaces[[vf2]]][Take[f2,-1][[1]]],

```

```
Flatten[Map[dots2[d,w,k][visiblefaces[[vf1+1]]],  
commence[d,w,k,n][visiblefaces[[vf1+1]]],1]],  
  
If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]],  
Join[dots[d,w,k][visiblefaces[[1]]][x],  
dots3[d,w,k][visiblefaces[[1]]][Take[dots[d,w,k]  
[visiblefaces[[1]]][x],-1][[1]]],  
Flatten[Map[dots2[d,w,k][visiblefaces[[1]]],  
commence[d,w,k,n][visiblefaces[[1]]],1]],  
  
If[nvisiblefaces==3&&ju[[2]]<=k<=ku[[2]],  
Join[dots[d,w,k][visiblefaces[[2]]][x],  
dots3[d,w,k][visiblefaces[[2]]][Take[dots[d,w,k]  
[visiblefaces[[2]]][x],-1][[1]]],  
Flatten[Map[dots2[d,w,k][visiblefaces[[2]]],  
commence[d,w,k,n][visiblefaces[[2]]],1]],  
  
If[nvisiblefaces==3,  
Join[dots[d,w,k][visiblefaces[[3]]][x],  
dots3[d,w,k][visiblefaces[[3]]][Take[dots[d,w,k]  
[visiblefaces[[3]]][x],-1][[1]]],  
Flatten[Map[dots2[d,w,k][visiblefaces[[3]]],  
commence[d,w,k,n][visiblefaces[[3]]],1]],  
  
If[nvisiblefaces==2&&ju[[1]]<=k<=ku[[1]]&&ju[[2]]<=k<=ku[[2]],  
(fd=Flatten[Map[firstdot[d,2.875,k],  
visiblefaces],1];  
vf1=Flatten[Position[fd,Min[fd]],1][[1]];  
vf2=Flatten[Position[fd,Max[fd]],1][[1]];  
Join[f1=dots[d,w,k][visiblefaces[[vf1]]][x],  
f2=dots[d,w,k][visiblefaces[[vf2]]][Take[f1,-1][[1]]],  
dots3[d,w,k][visiblefaces[[vf2]]][Take[f2,-1][[1]]],  
Flatten[Map[dots2[d,w,k][visiblefaces[[vf1]]],  
commence[d,w,k,n][visiblefaces[[vf1]]],1]],  
  
If[nvisiblefaces==2&&ju[[1]]<=k<=ku[[1]],  
Join[dots[d,w,k][visiblefaces[[1]]][x],  
dots3[d,w,k][visiblefaces[[1]]][Take[dots[d,w,k]  
[visiblefaces[[1]]][x],-1][[1]]],  
Flatten[Map[dots2[d,w,k][visiblefaces[[1]]],  
commence[d,w,k,n][visiblefaces[[1]]],1]],  
  
If[nvisiblefaces==2&&ju[[2]]<=k<=ku[[2]],  
Join[dots[d,w,k][visiblefaces[[2]]][x],  
dots3[d,w,k][visiblefaces[[2]]][Take[dots[d,w,k]  
[visiblefaces[[2]]][x],-1][[1]]],  
Flatten[Map[dots2[d,w,k][visiblefaces[[2]]],  
commence[d,w,k,n][visiblefaces[[2]]],1]],  
Join[dots[d,w,k][visiblefaces[[1]]][x],  
dots3[d,w,k][visiblefaces[[1]]][Take[dots[d,w,k]  
[visiblefaces[[1]]][x],-1][[1]]],  
Flatten[Map[dots2[d,w,k][visiblefaces[[1]]],  
commence[d,w,k,n][visiblefaces[[1]]],1]]]]]]]]]]]]]]]
```

commence gives an interval of x co-ordinates for dots by choosing seven pseudorandom real numbers between the x co-ordinate of one dot and the x-co-ordinate of its matching dot to the right for the appropriate face of the cube. The random number generator is seeded with the integer n so that it is possible to obtain the same sequence of numbers in a repeat run of the program. The given dot is included in the interval here.

```
commence[d_,w_,k_,n_] [List[a_,b_,c_,g_]]:=
(SeedRandom[n+Ceiling[5*k]];intval:=Table[Random[Real,
{firstdot[d,w,k][List[a,b,c,g]][[1]],
mainfun[d,w,k][List[a,b,c,g]][firstdot[d,w,k]
[List[a,b,c,g]][[1]]]],
{7}]/N;
Join[firstdot[d,w,k][List[a,b,c,g]],intval])
```

pts again considers many possible cases in order that the correct **commence** is chosen . i.e. we again need to know the correct face order for each row of dots i.e. for each k. Once the appropriate x-co-ordinates have been found they are appended by k (a z co-ord.) to give a set of points in the x-z plane.

```
pts[d_,w_,n_] [visiblefaces_] [k_]:=
(If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]]&&ju[[2]]<=k<=ku[[2]]&&ju[[3]]<=k<=ku[[3]]
(fD=Flatten[Map[firstdot[d,2.875,k],visiblefaces],1];
vf1=Flatten[Position[fD,Min[fD]],1][[1]];
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[[vf1]]],1)),
If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]]&&ju[[2]]<=k<=ku[[2]],
( fD=Flatten[Map[firstdot[d,2.875,k],
Drop[visiblefaces,{3,3}],1];
vf1=Flatten[Position[fD,Min[fD]],1][[1]];
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[[vf1]]],1)),
If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]]&&ju[[3]]<=k<=ku[[3]],
( fD=Flatten[Map[firstdot[d,2.875,k],
Drop[visiblefaces,{2,2}],1];
vf1=Flatten[Position[Flatten[Map[firstdot[d,2.875,k],visiblefaces],1],
Min[fD]],1][[1]];
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[[vf1]]],1)),
If[nvisiblefaces==3&&ju[[2]]<=k<=ku[[2]]&&ju[[3]]<=k<=ku[[3]],
( fD=Flatten[Map[firstdot[d,2.875,k],
Drop[visiblefaces,{1,1}],1];
vf1=Flatten[Position[fD,Min[fD]],1][[1]];
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[[vf1+1]]],1)),
If[nvisiblefaces==3&&ju[[1]]<=k<=ku[[1]],
(vf1=1;
spots=Flatten[
Map[cuberowdots[d,w,k,n],
```

```

commence[d,w,k,n][visiblefaces[{{vf1}}],1)),
If[nvisiblefaces==3&&ju[[2]]<=k<=ku[[2]],
(vf1=2;
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[{{vf1}}],1)),
If[nvisiblefaces==3,
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[{{3}}],1],

If[nvisiblefaces==2&&ju[[1]]<=k<=ku[[1]]&&ju[[2]]<=k<=ku[[2]],
( fd=Flatten[Map[firstdot[d,2.875,k],visiblefaces],1];
vf1=Flatten[Position[fd,Min[fd]],1][[1]];
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[{{vf1}}],1)),
If[nvisiblefaces==2&&ju[[1]]<=k<=ku[[1]],
(vf1=1;
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[{{vf1}}],1)),
If[nvisiblefaces==2&&ju[[2]]<=k<=ku[[2]],
(vf1=2;
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[{{vf1}}],1)),
spots=Flatten[
Map[cuberowdots[d,w,k,n],
commence[d,w,k,n][visiblefaces[{{1}}],1]]]]]]]]];
Map[Append[#,k]&,Partition[spots,1]];

```

stereog concatenates all the points row by row for the stereogram by looking at each value of k and then plots them. There are two possible versions of stereog following. The first returns a black and white stereogram, while the second contains a colour function to enable colour changes depending on the row number of the stereogram.

```

stereog[d_,w_,n_,steps_][visiblefaces_]:=
ListPlot[Join[Flatten[Table[pts[d,w,n][visiblefaces][k],
{k,Min[ju]+0.00006,Max[ku]-0.00006,steps}],1],
Flatten[Table[plptsr[d,w,k,n][List[a,b,c,g]],
{k,-10,Min[ju]-0.00006,steps}]
,1],Flatten[Table[plptsr[d,w,k,n][List[a,b,c,g]],
{k,Max[ku]+0.00006,10,steps}]
,1]],
Prolog->AbsolutePointSize[1],
Axes->False,
PlotRange->{{-10,10},{-10,10}},
AspectRatio->Automatic]

stereog[d,2.875,173,1/10][visiblefaces]

```

```

stereog[d_,w_,n_,steps_][visiblefaces_]:=
Show[Graphics[Join[Table[{Hue[Abs[k]/(Abs[k]+1)]},Map[Point,pts[d,w,n]
[visiblefaces][k]]],
{k,Min[ju]+0.00006,Max[ku]-0.00006,steps}],
Table[{Hue[Abs[k]/(Abs[k]+1)]},Map[Point,plptsr[d,w,k,n][List[a,b,c,g]
{k,-10,Min[ju]-0.00006,steps}]
,Table[{Hue[Abs[k]/(Abs[k]+1)]},Map[Point,plptsr[d,w,k,n][List[a,b,c,g]
{k,Max[ku]+0.00006,10,steps}]]],
Prolog->AbsolutePointSize[1],
Axes->False,
PlotRange->{{-10,10},{-10,10}},
AspectRatio->Automatic]]

stereog[d,2.875,173,1/10][visiblefaces]

```

The following program gives us a perspective drawing of our cube. It may be used to create an anaglyph of a cube by choosing the two different viewpoints of our eyes. It needs to be amended if only the visible faces are to be drawn as currently some unwanted lines are drawn, as it is designed for the whole cube.

```

<<Graphics`ImplicitPlot`

ObjectPlane[xpt_List]:=xpt[[2]]-30;
ViewPt:={0,0,0}

ViewLine[xpt_List,t_]:=ViewPt+t*(xpt-ViewPt);

SurfaceIntersectionList[xpt_List]:=t /.
Solve[ObjectPlane[ViewLine[xpt,t]]==0,t];

PictureList[xpt_List]:=Map[ViewLine[xpt,#]&,
SurfaceIntersectionList[xpt]];

```

Note: At this point it must be decided which objcube is required.

```

objcube:=Map[PictureList,cornerRadius]/N
objcube:=Map[PictureList,Flatten[visiblefaces,1]]/N
objcube1:=Flatten[objcube,1]
v:=objcube1[[{1,2,3,4,5,6,7,8},{1,3}]]
edges:={v[[{1,5}]],v[[{1,4}]],v[[{1,2}]],v[[{2,6}]],
v[[{2,3}]],v[[{3,4}]],v[[{3,7}]],v[[{4,8}]],v[[{5,8}]],
v[[{5,6}]],v[[{6,7}]],v[[{8,7}]]}
lines:=Map[Line,edges]/N

```

```
Show[Graphics[{RGBColor[0,0.35,0],AbsoluteThickness[1],lines},Axes->False,  
PlotRange->{{-10,10},{-10,10}},  
PlotRegion->{{0,1},{0,1}},  
AspectRatio->Automatic]]
```

-Graphics-

```
Show[Graphics[{AbsoluteThickness[1],lines},Axes->False,  
PlotRange->{{-10,10},{-10,10}},  
PlotRegion->{{0,1},{0,1}},  
AspectRatio->Automatic]]
```

-Graphics-

```
Show[Graphics[lines,Axes->False,  
PlotRange->{{-10,10},{-10,10}},  
PlotRegion->{{0,1},{0,1}},  
AspectRatio->Automatic]]
```

-Graphics-

```
Show[Graphics[{RGBColor[0,0.35,0],lines}],Axes->True,  
PlotRange->{{-10,10},{-10,10}},  
PlotRegion->{{0,1},{0,1}},  
AspectRatio->Automatic]
```

-Graphics-

Appendix G

Program for creating Single-Image Stereogram of a special sphere

- This program returns a Single Image Stereogram (SIS), of a sphere which remains in a fixed position when it is viewed with either the 'crossed' or 'uncrossed' method. When viewed with the uncrossed method the inside of the sphere is seen behind the page of the SIS and when viewed with the crossed method the outside front of the same sphere is perceived to be 'floating' in front of our page (or above if the page is flat on a table and the viewer is looking down on it from above) between the viewer and SIS.

?circlei

circlei[tangentpoint, distance][z co-ord.of the line where the viewer's eyes meet the SIS] gives the centre of the circle which has a given tangent point on the x-axis with the associated tangent passing through the midpoint of the viewer's eyes. The viewer is viewing from a point which is a distance d in front of the x-axis (y co-ord=-d) and the viewline from the eyes meets the SIS at a height z=i.

```
circlei[q_,d_][i_]:=
  s=x/.Solve[x^2+i^2==q^2,x]//N;
  t=Min[s];
  u=Max[s];
  d'=-Max[Sqrt[i^2+d^2]]//N;
  slopel1=(d'-0)/(0-t);
  sloperadt=-1/slopel1;
  slopel2=(d'-0)/(0-u);
  sloperadu=-1/slopel2;

  centre=Flatten[Solve[{b/(a-t)==sloperadt,
    b/(a-u)==sloperadu},{a,b}]]//N;
  { centrex=a/.centre,
    centrey=b/.centre})
```

radius returns the radius of the required circle.

```
radius[q_,d_][j_]:=
  (cj=circlei[q,d][j];
  ((cj[[1]]-u)^2+cj[[2]]^2)^(1/2))
```

?start

start[tangent point x co-ord., y-co-ord. of midpoint of viewer's eyes, half the eye separation, z co-ord. of the line where the viewlines meet the SIS] gives a random interval of points xj on the x-axis calculated using the point where the circle $x^2+y^2=(\text{tangent point } x)^2$ cuts the positive z-axis, and the line from this point through the viewer's left eye. This line cuts the x-axis at a point p. A specified number of random points, say 2, are chosen between 0 and p and then the mirror images of each of these points about the y-axis are chosen. The interval consists of 6 points in this case (includes the points p and -p). Their x co-ords are listed.


```

start[q_,d_,w_,j_] := (cj=circlei[q,d][j];
  p:=x/.Solve[(-(cj[[2]]+d'+
  ((cj[[1]]-u)^2+cj[[2]]^2)^(1/2))/w)*x+
  cj[[2]]+((cj[[1]]-u)^2+
  cj[[2]]^2)^(1/2)==0,x]/N;

  {int1=Table[Random[Real,{0,Max[p]}],{4}],
  int2=Map[mainfun[q,d,w,j],int1]}/N)

```

(hilbert) In[55]:=

?mainfun

mainfun[tangent point x co-ord.,y-co-ord. of midpoint of viewer's eyes, half the eye separation,z co-ord.of the line where the viewlines meet the SIS]
maps a point on the x-axis onto its corresponding point on the x-axis and we are moving from left to right.

```

mainfun[q_,d_,w_,j_][Intr_]:=
(c=x/.Solve[(x-circlei[q,d][j][[1]])^2+
  ((-d/(Intr+w))*(x-Intr)-
  circlei[q,d][j][[2]])^2-((circlei[q,d][j][[1]]-u)^2+
  (circlei[q,d][j][[2]]-0)^2)==0,x];
If[(-d/(Intr+w))<0,f=-Max[Map[Abs,c]],f=Max[c]];
ycoord=y/.Solve[(f-circlei[q,d][j][[1]])^2+
  (y-circlei[q,d][j][[2]])^2-
  ((circlei[q,d][j][[1]]-u)^2+(circlei[q,d][j][[2]]-0)^2)
  ==0,y];
circlePt={f,Max[ycoord]};
slopeR=(d-circlePt[[2]])/(w-circlePt[[1]])/N;
x/.Solve[slopeR*(x-w)+d==0,x] // First);

```

(hilbert) In[57]:=

?dots

dots[tangent point x co-ord.,y-co-ord. of midpoint of viewer's eyes, half the eye separation, z co-ord.of the line where the viewlines meet the SIS] is an iterative function applied to the x-co-ord of a point which recursively uses mainfun to obtain corresponding points on the x-axis until the given condition applies.

```

dots[q_,d_,w_,j_][c_]:=
  Drop[FixedPointList[mainfun[q,d,w,j],c,
    SameTest ->((#1-#2)>0&)],-1];

```

Note: Below I have replaced start[q,d,w,j][[1]] by int1 and start[q,d,w,j][[2]] by int2.

?points

points[tangent point x co-ord.,y-co-ord. of midpoint of viewer's eyes, half the eye separation] [z co-ord. of the line where the viewlines meet the SIS] maps dots onto each point of the starting interval to give a list of x values to the right of our starting interval. To obtain a full row of x values for our dots we take the Union of this list with their mirror images about the y-axis and with the points in the starting interval. This list of x-co-ords. is appended by the appropriate second co-ordinate which equals the height of the row above or below the x-axis.

```
points[q_,d_,w_][j_]:= (spots:=Join[start[q,d,w,j][[2]],
Flatten[Map[dots[q,d,w,j],int2],1],
int1,
-1*Flatten[Map[dots[q,d,w,j],
-1*int1],1]]];

Map[Append[#,j]& ,Partition[spots,1]]];
```

□ Note: Below we have three possible versions of the function suitable to return our SIS.

The first case allows the plotting of geometrical shapes instead of just dots at each point. There is a problem with regulating the shapes' size. This may be overcome with extra study of graphics.

The second just returns the SIS with dots of an absolute size.

The third allows the introduction of variable colour by introducing the Hue function whose argument changes with each row of dots.

```
(hilbert) In[59]:=
```

?stereog

```
stereog[tangent point x co-ord.,y-co-ord. of midpoint of viewer's
eyes, half the eye separation, minimum row z-value, maximum row
z-value,step size as we move from one row to the next]
returns our SIS with the appropriate options taken into
consideration. Correct plot range for the size of our sphere is
vital to have the correct scale.
Eye-spacing does not vary.
```

Only one of the following stereog functions must be entered.

```

stereog[q_, d_, w_, zmin_, zmax_, steps_] :=
(square=RegularPolygon[4, 0.003, {0, 0},
Pi/4];
MultipleListPlot[Flatten[Table[points[q, d, w][j],
{j, zmin, zmax, steps}], 1],
DotShapes->{MakeSymbol[square]}, Axes->False,
PlotRange->{{-10, 10}, {-10, 10}}, AspectRatio->Automatic]]

```

To obtain our stereogram after we have entered all the preceding functions, we enter the following with our chosen values as arguments.

```
stereog[8, -30, 2.875, -7.8, 7.8, 2/10]
```

Following we have our alternative stereog functions.

```

stereog[q_, d_, w_, zmin_, zmax_, steps_] :=

ListPlot[Flatten[Table[points[q, d, w][j],
{j, zmin, zmax, steps}], 1],
Prolog->AbsolutePointSize[1], Axes->False,
PlotRange->{{-10, 10}, {-10, 10}}, AspectRatio->Automatic]

stereog[q_, d_, w_, zmin_, zmax_, steps_] :=
Show[Graphics[Table[
{Hue[Abs[j]/(Abs[j]+1)]}, Map[Point, points[q, d, w][j]],
{j, zmin, zmax, steps}],
Prolog->AbsolutePointSize[1], Axes->False,
PlotRange->{{-10, 10}, {-10, 10}}, AspectRatio->Automatic]]

stereog[q_, d_, w_, zmin_, zmax_, steps_] :=
Show[Graphics[Table[
{Hue[Abs[j]/Ceiling[(Abs[j]+1)]]}, Map[Point, points[q, d, w][j]],
{j, zmin, zmax, steps}],
Prolog->AbsolutePointSize[0.1], Axes->False,
PlotRange->{{-10, 10}, {-10, 10}}, AspectRatio->Automatic]]

```

Bibliography

- [1] Dror Bar-Natan. Random-Dot Stereograms. *Mathematica*, 1(3):69–75, 1991.
- [2] R.M. Burde and S.E. Feldon. The Extraocular Muscles. In Hart [10], pages 101–183.
- [3] Alison Cole. *Perspective: A Visual guide to the theory and techniques from the Renaissance to Pop Art*. HarperCollins, Pymble, Australia, 1993.
- [4] R. Courant and H. Robbins. *What is Mathematics?* Oxford University Press, New York, 1958.
- [5] N.E. Thing Enterprises. *Magic Eye II*. Viking, Penguin Books, Victoria, Australia, 1994.
- [6] M. Gardner. The Curious Magic of Anamorphic Art. *Scientific American*, pages 110–116, 1975.
- [7] Richard L. Gregory. *The Intelligent Eye*. Weidenfeld and Nicolson, London, 1970.
- [8] W.E.L. Grimson. *From Images to Surfaces*. Massachusetts Institute of Technology, U.S.A, 1981.
- [9] H. Hamngren. My Anamorphoses: Types that produce three kinds of images in circular cylindrical mirrors. *Leonardo*, 14(3):198–201, 1981.
- [10] William M. Hart, editor. *Adler's Physiology of the Eye*, volume Ninth. George S. Stamathis, St. Louis, Missouri, 1992.
- [11] P. Hickin. Anamorphosis. *The Mathematical Gazette*, 76(476):209–221, 1992.

- [12] D. H. Hubel. *Eye, Brain, and Vision*, volume 22. Scientific American Library, New York, 1988.
- [13] B. Julesz. *Foundations of Cyclopean Perception*. University of Chicago Press, Chicago, 1971.
- [14] P. W. Kuchel. Anamorphoses: A Visual Aid for Circle Inversion. *The Mathematical Gazette*, 63:82–89, 1979.
- [15] F. Leeman. *Hidden Images*. Harry N. Abrams, New York, 1976.
- [16] R. K. Luneburg. *Mathematical Analysis of Binocular Vision*. Hanover Institute, Princeton, New Jersey, 1947.
- [17] R. E. Maeder. Single-Image Stereograms. *The Mathematica Journal*, 5(1):50–61, 1995.
- [18] J. McGregor and A. Watt. *The Art of Graphics for the IBM PC*. Addison-Wesley, Wokingham, England, 1986.
- [19] S. P. McKee and G. J. Mitchison. The Role of Retinal Correspondence in Stereoscopic Matching. *Vision Research*, 28(9):1001–1012, 1988.
- [20] L. Tychsen, M.D. Binocular Vision. In Hart [10], pages 773–853.
- [21] G. J. Mitchison and G. Westheimer. The Perception of Depth in simple Figures. *Vision Research*, 24(9):1063–1073, 1984.
- [22] D. Pedoe. *An Introduction to Projective Geometry*. Pergamon Press, New York, 1963.
- [23] T. P. Piantanida. Stereo Hysteresis Revisited. *Vision Research*, 26(3):431–437, 1986.
- [24] H. Rheingold, editor. *Stereogram*. Angus and Robertson, Sydney, Australia, 1994.
- [25] I. Rock. *Perception*. Scientific American Library, New York, 1995.
- [26] D. H. Row. Geometry and our World. Lecture notes University of Tasmania, 1994.
- [27] M. S. Terrell and R. E. Terrell. Behind the Scenes of a Random Dot Stereogram. *American Mathematical Monthly*, 101(8):715–724, 1994.

- [28] C. W. Tyler and M. B. Clarke. The Autostereogram. *SPIE*, 1256:182–197, 1990.
- [29] O. Veblen and J. W. Young. *Projective Geometry*, volume 1 & 2. Blaisdell, London, 1938.
- [30] A.F. Walonker and S.E. Feldon. Clinical Assessment of binocular vision. In Hart [10], pages 183–197.

Index

Index

- accommodation, 34
- acuity, 34
- anaglyph, 27
- anamorphogram, 97
- autostereogram, 29
- axis of central collineation, 194
- axis of perspectivity, 184
- background plane, 150
- central axis, 86
- central collineation, 194
- central involution, 194
- centre of central collineation, 194
- centre of perspectivity, 183
- circuit; 4-pt, 197
- closest vertex, 163
- collineation, 194
- conic, 185
- conical mirror, 122
- conical picture surface, 114
- correct picture, 108
- corresponding points, 37
- crossed disparity, 43
- crossed-eyes viewing technique, 56
- cube stereogram construction, 156
- cube; parametric representation, 104
- Cyclopean eye, 29
- cylindrical mirror, 119
- cylindrical picture surface, 111
- depth, 47
- diagonal points, 197
- dim, 104
- disparate, 50, 87
- disparity, 37, 41
- distance, 47
- distance of the viewer from a stereogram, 13
- Dot, 161
- double point, 45
- dual, 177
- dual viewing, 177
- elation, 194
- elementary map, 181
- ellipsoid stereogram construction, 144
- equally-spaced dots, 61
- extended Euclidean plane, 177
- eye-spacing ranges, 59
- far perspective image, 111
- far point, 43
- figure, 98
- fixate, 35
- fixation line, 76
- fixation plane, 76
- fixation point, 41
- fixed sphere, 177
- fovea, 34
- foveola, 34
- fusion, 35
- generating bases of a conic, 186
- grid; parametric representation, 99
- guiding dots, 64

harmonic conjugate, 180
 homogeneous co-ordinates, 104
 homologous elements, 181
 homology, 194
 horopter, 38
 horopter curvers, 89

 ideal line, 176
 ideal point, 176
 image, 98
 image figure, 100
 inverse image in stereoscope, 180

 line of boundary, 160
 Luneburg's experiments, 89

 matching elements, 49
 microsaccades of eyes, 53
 mirror anamorphograms, 115
 monocular vision, 98

 near perspective image, 111
 near point, 43
 nodal point, 34
 non-singular conic, 186

 Panum's area, 50
 Panum's fusional space, 50
 parallel viewer, 58
 parallel viewing technique, 55
 pencil of lines, 182
 perceived dots, number, 62
 perspective, 23, 182
 perspective drawing, 98
 perspectivity, 183, 184
 picture plane, 23
 Principle of Duality, 177
 projection, 181
 projective plane, 177

 projectivity, 184

 random-dot stereogram, 28
 range of points, 182
 receptive field, 51

 saccadic eye movement, 52
 Single-Image Stereogram, 29
 Single-Image Stereogram construction, 135
 singular conic, 186
 slant anamorphogram, 25
 spherical mirror, 125
 starting interval, 138
 stereogram, 27
 stereopsis, 26
 stereoscope, 26
 stereoscope, geometric, 178
 stereoscope, geometric inverse, 178
 sub-pencil, 182

 translation, 193

 uncrossed disparity, 43
 uncrossed viewing technique, 55
 uniocular projection, 59

 vergence, 35
 vertex of pencil of lines, 182
 Vieth-Muller circle, 37
 viewing techniques, stereograms, 55
 viewline, 34, 59
 visible faces, 163
 visible faces of cube, 156
 visual axis, 35
 visual field, 34

 Wall-paper Effect theorem, 68
 wallpaper effect, 29

 zero disparity, 50, 87